Mission Research Corporation

MRC/WDC-R-316

DESIGN AND TESTING OF UBITRON AMPLIFIER TUBES

Dean Pershing

October 1993

Final Technical Report

Prepared for: Naval Research Laboratory
4555 Overlook Avenue, S.W.
Washington, D.C. 20375

Contract No. N00014-85-C-2141

Prepared by: MISSION RESEARCH CORPORATION
8560 Cinderbed Road, Suite 700
Newington, Virginia 22122
(703) 339-6500

MRC/WDC-R-316

Statement A per telecon
Dr. Robert Jackson ONR/Code 6843
Washington, DC 20375

Copy 18

NWW 12/2/93

DESIGN AND TESTING OF UBITRON AMPLIFIER TUBES

Accesion For		
NTIS CRA&I	X	
DTIC TAB	☐	
Unannounced	☐	
Justification		
By		
Distribution /		
Availability Codes		
Dist	Avail and / or Special	
A-1		

Dean Pershing

October 1993

<u>Final Technical Report</u>

Prepared for: Naval Research Laboratory
4555 Overlook Avenue, S.W.
Washington, D.C. 20375

Contract No. N00014-85-C-2141

Prepared by: MISSION RESEARCH CORPORATION
8560 Cinderbed Road, Suite 700
Newington, Virginia 22122
(703) 339-6500

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

SECTION 1
INTRODUCTION

Within the past decade, free-electron lasers (FELs) have received considerable attention because of their potential as high power, tunable sources of coherent radiation. In particular, experiments employing high current beams at low voltages show promise as millimeter and submillimeter wavelength sources. The development of low-voltage, high-current FELs can be traced to experiments at General Electric conducted by R. M. Phillips in the late 1950's and early 1960's on a device which he called a ubitron [1,2,3]. Table 1.1 summarizes the results of Phillips' experiments along with results from some recent high current FEL amplifier experiments and the design values of the present experiment. Although some aspects of ubitron performance have been surpassed by recent experiments, the original ubitron results are still impressive. This fact, combined with advances from theoretical and experimental research on collective free-electron lasers, indicate that the ubitron has great potential as a high-power, micro- and millimeter wave source.

An experimental program to study the potential of the ubitron interaction mechanism is underway at the Naval Research Laboratory. Key developments for improved performance include the use of circularly polarized magnetic wiggler and waveguide fields, high quality electron beams, gyroresonant enhancement of the wiggler field, and enhancement through system parameter tapering. An amplifier experiment has been constructed at NRL to explore the effects of these developments on the gain and efficiency limits of the ubitron. The goals of the experiment are to demonstrate that the ubitron is a high-gain, high-efficiency, wide-bandwidth radiation source. Specifically, this experiment is expected to realize a total gain between 25 and 30 dB at an efficiency greater than 10%, and an instantaneous bandwidth exceeding 20% at a center frequency of 14.5 GHz.

The single-state, single-pass amplifier configuration of the present ubitron experiment has been designed to match, as closely as possible, the assumptions of an NRL-The single-stage, single-pass amplifier configuration of the present ubitron experiment developed 3-D ubitron theory [4, 5]. This configuration utilizes a high-quality solid cylindrical electron beam inside a smooth-wall

Table 1.1 Relevant Experimental Results.

	UBITRON					FEL	
FREQUENCY (GHz)	2.7	2.7*	15.7	54.1	14.5	34.6	35
PEAK POWER (MW)	1.2	1.2	1.65	.154	(1-5)	180	17
EFFICIENCY (%)	10	13	6	6.2	(15-20)	7	3
TOTAL GAIN (dB)	13	15	-	30	(30)	50	50
BANDWIDTH (%)	30	30	-	-	(20-30)	≈10	-
MODE	TE_{10}	TE_{10}	TE^{o}_{01}	TE^{o}_{01}	TE^{o}_{11}	TE_{01}	TE^{o}_{11}
VOLTAGE (kV)	125	125	200	67	250	3300	900
CURRENT (A)	64	64	125	37	30-100	850	600
GUIDE FIELD (kG)	0	0	0	10	1-6.5	0	11.75
LABORATORY	GE	GE	GE	GE	NRL	LLNL	NRL

* - TAPERED INTERACTION PARAMETERS

( ) - CALCULATED PERFORMANCE

circular waveguide that is confined by a uniform axial magnetic field. The interaction relies on the coupling between the LHCP TE_{11} RF mode and the wiggler shifted beam negative energy space charge wave, i.e. the device is designed to operate in the collective FEL mode. The wiggler field is generated by a double taper RHCP bifilar helical electromagnet. A comparison of the experimentally measured small and large signal gain, efficiency, and bandwidth with theoretical predictions will allow a meaningful assessment of the ubitron as a viable high-power source.

1.1 GOALS AND PARAMETERS.

Recent experiments have made significant advances in several areas of FEL performance, especially in the areas of peak power and efficiency. However, there remain a number of issues which require investigation and which the present experiment has been designed to address. In particular, issues such as gain per free-space wavelength and instantaneous bandwidth are important parameters for many of the applications of ubitrons. Further improvements in efficiency without tapering are needed as well as a more detailed investigation of the potential of the wiggler/guide-field gyroresonance for performance enhancement. In designing the experiment, consideration was also given to matching the model employed in the 3-D ubitron theory in order to achieve a more direct comparison with the theory.

1.2 CALCULATED PERFORMANCE.

The major quantities that determine ubitron performance are gain, bandwidth, and efficiency. Since the gain is a generally increasing function of the ratio of transverse to axial beam velocity (alpha), high transverse velocities are required for high gain operation. The wiggler strength required to generate a given velocity ratio is a function of beam voltage, wiggler period, and axial guide field (due to the gyroresonance effect of combined helical wiggler and axial guide magnetic fields). Calculations of the small signal gain are shown in Fig. 1.1 for several values of alpha. The curve at alpha = 0.29 corresponds roughly to the design point of the NRL ubitron experiment. Note that the peak gain when plotted versus alpha lies along a straight line which passes through the origin. This is the expected dependence for a collective FEL interaction. The bandwidth

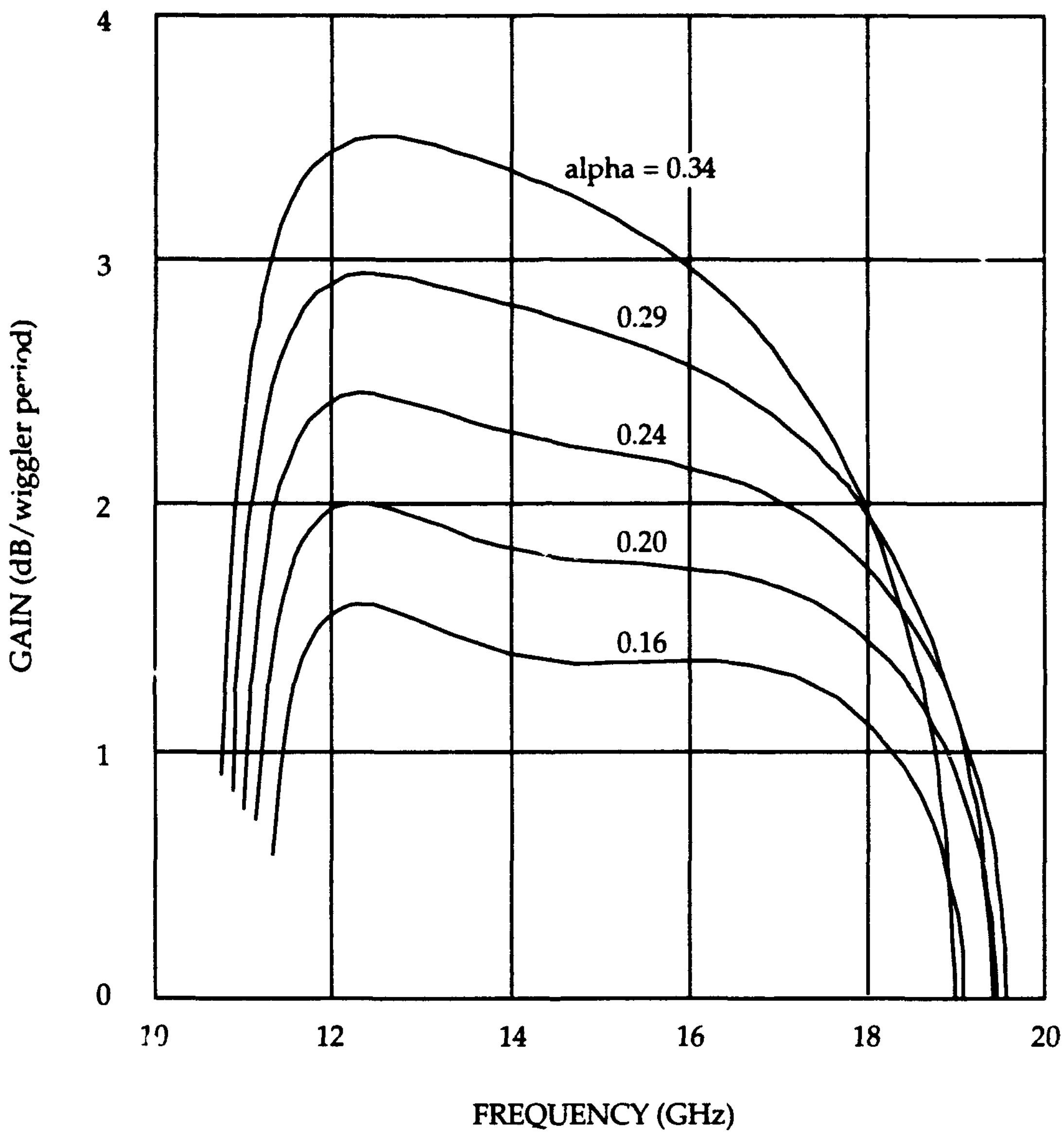


Figure 1.1. Small signal gain calculations.

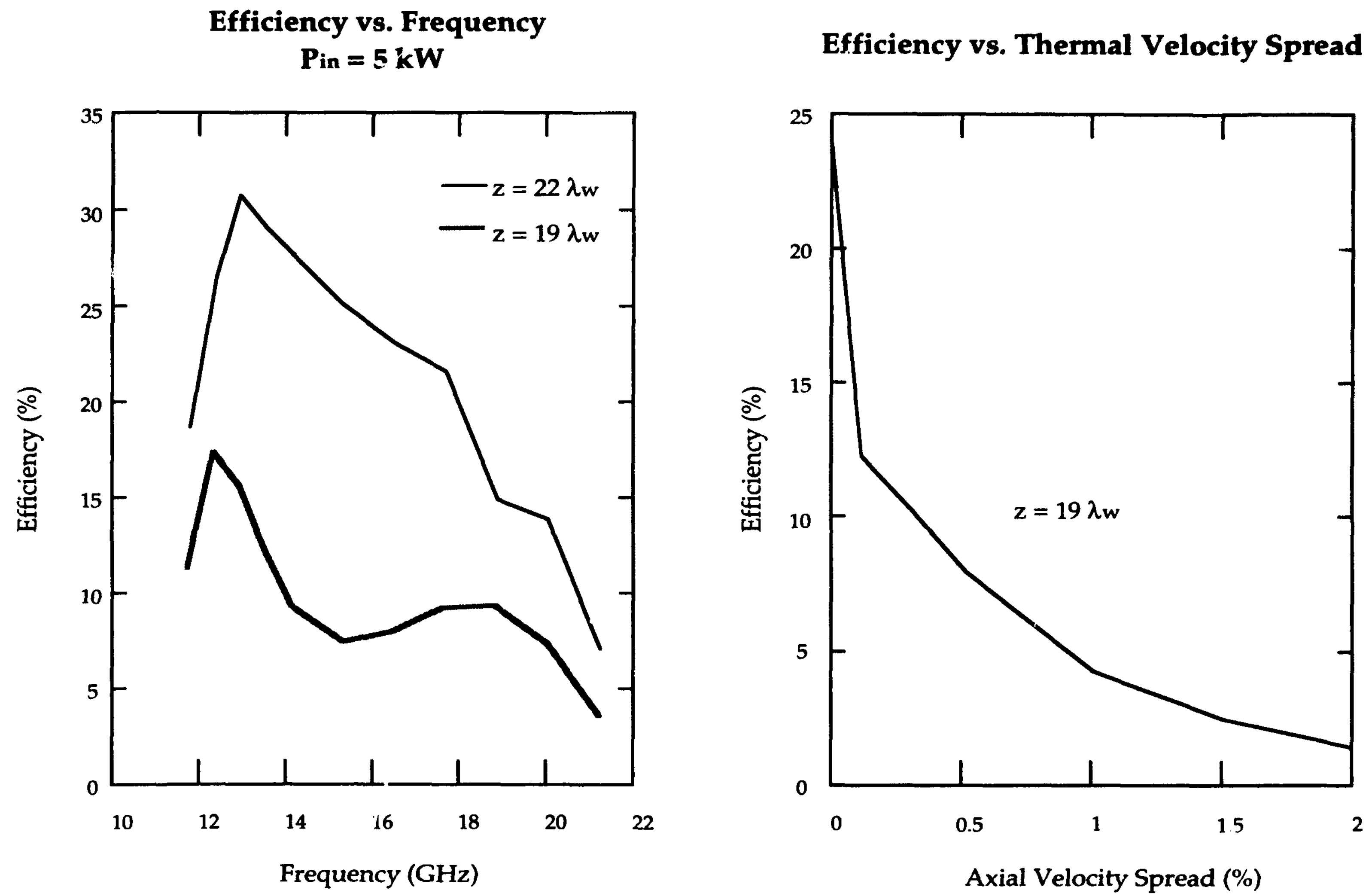
predicted by the small signal calculations is quite large and yields a predicted saturated bandwidth (assuming 30 dB total gain) of ≥ 20%. These calculations utilized a 3-D small signal code which was kindly provided to us by Drs. Freund and Ganguly [Ref. 4].

Recently, a 3-D nonlinear ubitron/FEL code has been developed at NRL [Ref. 6]. The nonlinear performance of the ubitron calculated with this code is shown in Fig. 1.2. The calculations which are shown in the figure include AC space-charge effects but do not include beam velocity spread, although the code does have the capability to include velocity spread. The peak efficiencies calculated are as high as 30-35% for an untapered configuration. Finite beam temperature of the degree produced in the 100-A electron gun should result in a reduction of peak gain by a factor of approximately one half [Ref. 7]. The capability of tapering various experimental parameters, eg. axial guide field, wiggler amplitude, wiggler period, etc., has been designed into the experiment to permit evaluation of these techniques for increasing the base efficiency.

1.3 EXPERIMENTAL CONFIGURATION.

The experiment has been designed with the following issues in mind: 1) generation of a high-quality electron beam, 2) generation of sufficient beam transverse velocity, and 3) high interaction impedance without oscillations or reflections. Beam quality is of paramount importance, since the predicted efficiency is critically dependent on a low axial velocity spread. The three major sources of velocity spread are the electron gun optics, wiggler field gradients, and the RF mode field gradients. The choice of a helical wiggler field/fundamental circular waveguide mode interaction minimizes the field gradient contribution to the velocity spread. The remaining major contributor to velocity spread, electron gun optics, can be minimized by careful gun design.

Since the gain is a generally increasing function of the transverse/axial velocity ratio, or alpha, high transverse velocities are required for high gain operation. A bifilar helix wiggler, with adiabatically tapered entrance and exit fields, is one method of generating the required wiggler strength. Although the design of such a wiggler is a formidable task, the following benefits accrue: 1) constant axial velocity orbits are attainable, 2) high beam quality in the wiggler



Vb = 250 kV, Ib = 100 A, Bz = 2 kG, Bw = 500 G, Lw = 2.54 cm

Figure 1.2. Nonlinear performance calculations for the NRL ubitron.

field, 3) continuous bunching force, and 4) results using this wiggler can be directly compared with theory.

The desired high interaction impedance can be achieved using the TE_{11} fundamental circular waveguide mode. Use of a smooth wall cylindrical waveguide will help to prevent spurious oscillations. To prevent reflections caused by an output window, the present experiment does not use one. Rather, the entire RF output is absorbed in a combination water load/calorimeter.

A schematic of the original NRL ubitron configuration is shown in Fig. 1.3. The major elements of the electron beam system are the electron gun, the drift tube/waveguide, and the water cooled beam collector. The three elements of the magnetics system are the solenoid, including trim coils, the double taper bifilar helix, and the kicker magnet for the collector. Major elements of the microwave system are driver amplifier(s), the input coupler to launch the LHCP TE_{11} wave, smooth wall cylindrical waveguide, and a water load/calorimeter to absorb the microwave radiation. Details of each major components' design and performance are presented in following sections.

Ku-BAND UBITRON AMPLIFIER

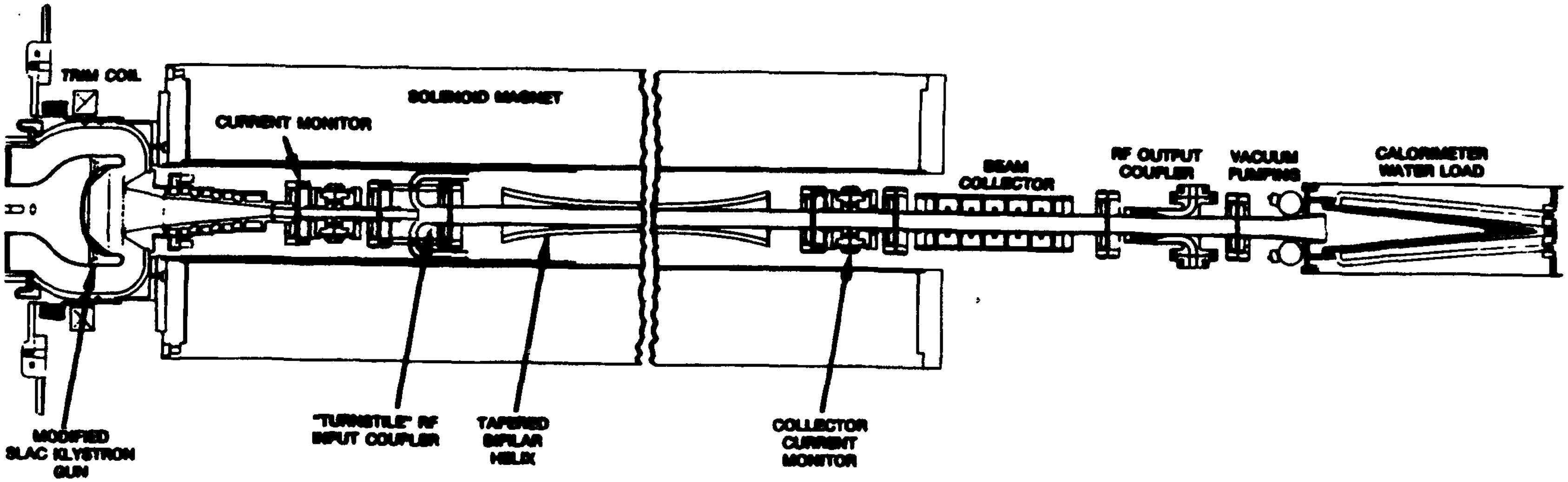


Figure 1.3. Experimental schematic for the NRL ubitron.

SECTION 2
COMPONENT DESCRIPTIONS

2.1 ELECTRON GUNS.

2.1.1 Introduction.

The interaction configuration chosen for the NRL ubitron utilizes a solid, uniform-density cylindrical electron beam propagating inside a hollow cylindrical drift tube/waveguide. Two electron guns of the 'Pierce' type are used to generate an electron beam of this geometry and with the desired low axial velocity spread. Although Pierce type electron guns are well known as high quality electron beam sources, very few have been designed to operate at voltages as high as 250 kV. Due to the technological challenges involved in this task, Varian Associates was contracted to design and fabricate two electron guns satisfying the specifications listed in Section 2.1.3.

2.1.2 SLAC Gun.

Since the remainder of the ubitron was expected to be finished before the electron guns, an interim gun, with possibly reduced beam specifications, was needed for initial ubitron testing. Surplus SLAC klystron guns were available and were used in this role. These guns were not, however, usable without modification. The pertinent parameters for these guns are: 250 kV, 250 A, 1 in. beam diameter in a 1 - 1.2 kG axial magnetic field. Both the emitted current and the beam diameter were too large for the ubitron. It was determined that it would not be practical to compress the beam magnetically to the desired 8-mm diameter and still maintain adequate beam quality.

Two major modifications were made to the SLAC gun in order to reduce both the emitted current and the beam diameter for use in the ubitron experiment. Current reduction was accomplished by a combination of reduced emission surface area and scraping of 'hot' beam edge electrons for further current reduction and selection of the low velocity spread central beam core. A reduction of beam diameter was also a result of beam scraping in which the anode was extended at half the slope of the original taper to the desired 8-mm

diameter. The scraper/trimmer region required additional water cooling to remove heat generated by collected electrons.

A reduction in emission surface area to reduce current required changes in the cathode shaping electrode geometry in order to maintain focused electron trajectories. The modified electrode compensated for the reduced space charge that would have been emitted from the outer part of the cathode in the original design. Further, the electrode was gold plated to inhibit any potential emission. A standard triple oxide (barium, strontium, calcium) coating was the electron emission source. While easy to apply, it is very susceptible to 'poisoning', or reduced emission caused by surface contamination. All gun modifications are shown in Figs. 2.1-2.

For initial ubitron operation, it was decided to locate the gun in the fringing field of the solenoid with pole piece, with some additional field shaping provided by a small (~200 G) trim coil over the cathode. The solenoidal field profile was calculated using POISSON [8, 9]. SCRIBE [10] simulations of this gun geometry were performed to find the relative axial positions of the cathode and solenoid that resulted in the best beam quality. For the case of the cathode 36.19 cm from the solenoid pole piece, simulations demonstrate good beam quality; total emitted current is reduced to ~80 A, of which 40-45 A pass through the beam tunnel with a normalized emittance of ~75 π-cm-rad. The emittance calculation probably includes primarily scraped edge particles, with a solenoidal field profile that was less than optimal. The resulting beam was deemed suitable for interim operation, although a truly laminar beam was not possible due to physical interference between the gun face and the solenoid pole piece when located in positions that result in better beam quality in simulations.

As mentioned above, this gun/solenoid configuration was used for initial ubitron operation, especially for the low voltage harmonic experiments. Harmonic operation required negative current at the trim coil which cancelled the magnetic field at the cathode, resulting in a highly rippled $p_\theta = 0$ beam when injected into a uniform magnetic field. This is discussed further in Section 3.1. A comparison of calculated rippled and laminar beam trajectories is shown in Fig. 2.3. See Section 2.3 for a comparison of rippled/laminar beam field profiles.

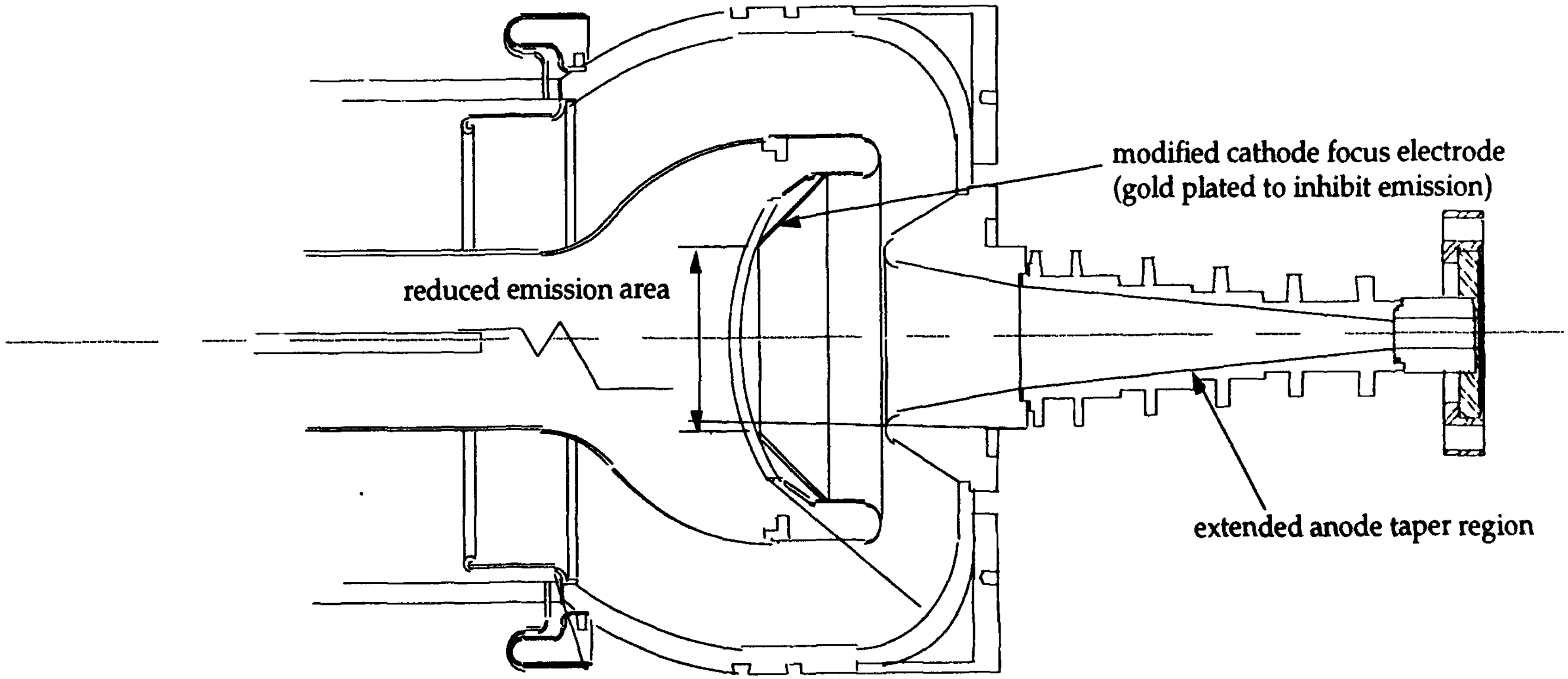


Figure 2.1. Construction details of the modified SLAC klystron gun.

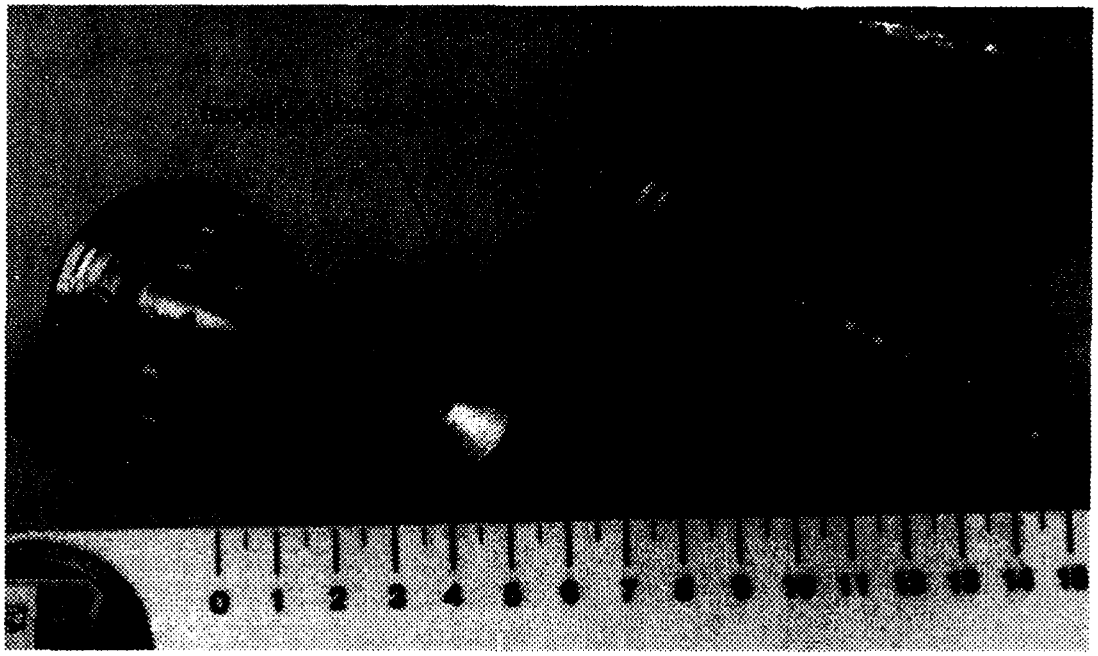


Figure 2.2. Modified SLAC klystron gun.

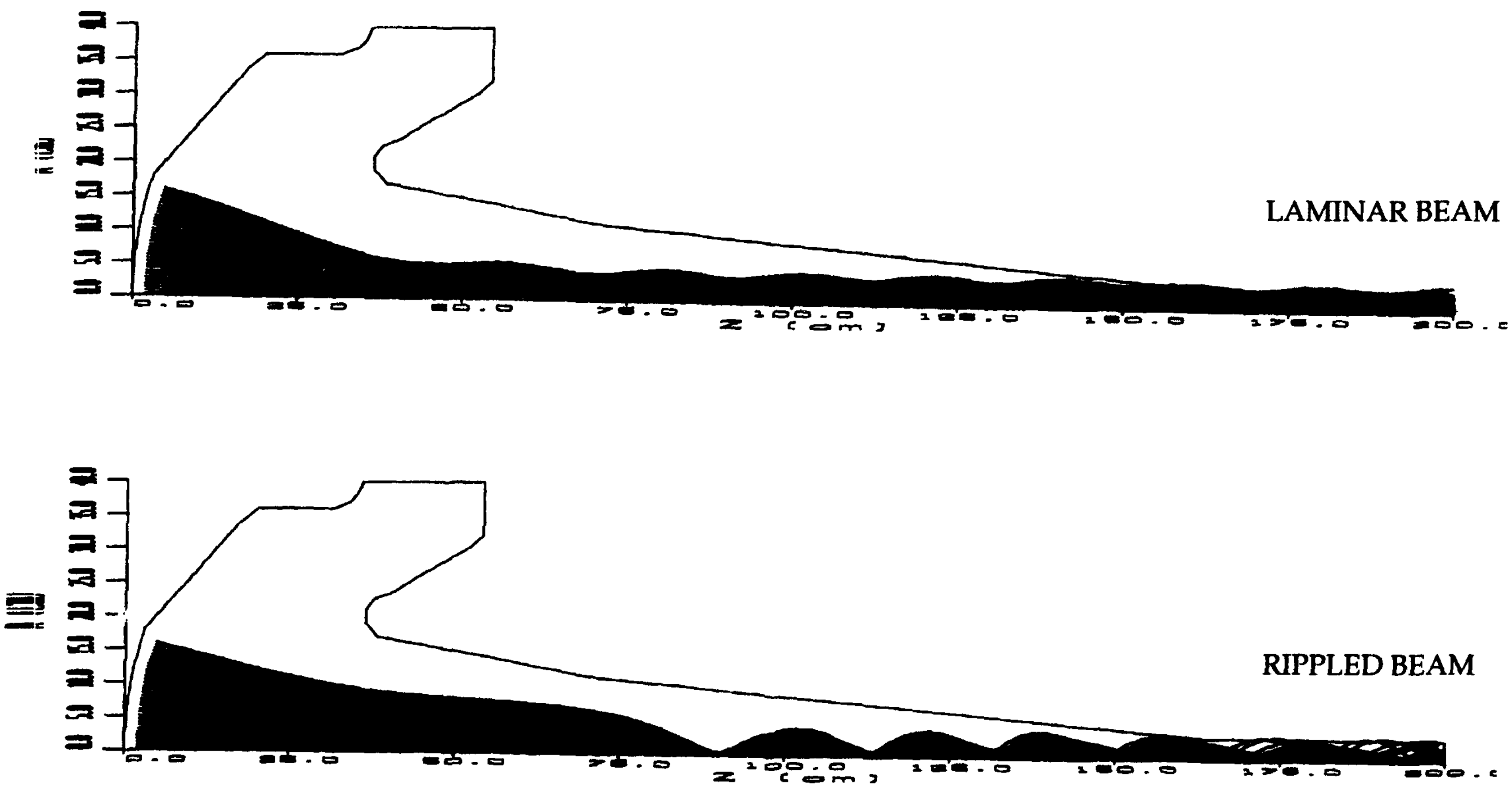


Figure 2.3. Comparison of rippled and laminar beam electron trajectories.

For normal high-voltage ubitron operation, however, a highly laminar beam is required. To achieve this with the SLAC gun, the solenoid pole piece was enlarged, allowing insertion of the gun further into the solenoid. Computed trajectories with this gun/solenoid configuration are shown in Fig. 2.4. The normalized emittance is found to be 2.8 π-cm-mrad with an axial velocity spread of 0.02 - 0.26%, depending on the axial field profile at the solenoid. Ubitron results presented in Section 3.2 were obtained using this gun/solenoid configuration. It should be noted that sustained gun operation above ~230 kV was not possible due to arcing.

2.1.3 Advanced Gun.

As previously discussed, ubitron operation is highly dependent on beam quality, in this context meaning low axial velocity spread. Varian Associates was contracted by NRL to design and build a Pierce-type electron gun with a confined-flow focusing system, satisfying specifications below. A diagram of this gun is shown in Fig. 2.5. [11].

Operating voltage (cathode pulse)	250 kV
Cathode current	100 A
Cathode heater voltage (max.)	30 VAC
Cathode heater current (max.)	25 A
Pulse length (max.)	2 μs
Repetition rate (max.)	100 Hz
Beam radius in 2.5 kG magnetic field	<0.4 cm
Beam centroid offset in 2.5 kG magnetic field	<0.005 cm
Beam axial velocity spread (computed) (biased standard deviation)	<0.4% with goal of 0.1% (~8.5% beam ripple)
Beam ripple (measured)	<20%
Concentricity of cathode and anode	±0.004 in.
Angular deviation (tilt) of cathode and anode relative to gun axis (max.)	0.005 radians
Lifetime (min.)	5000 hrs.
Capacitance (max.)	150 pF

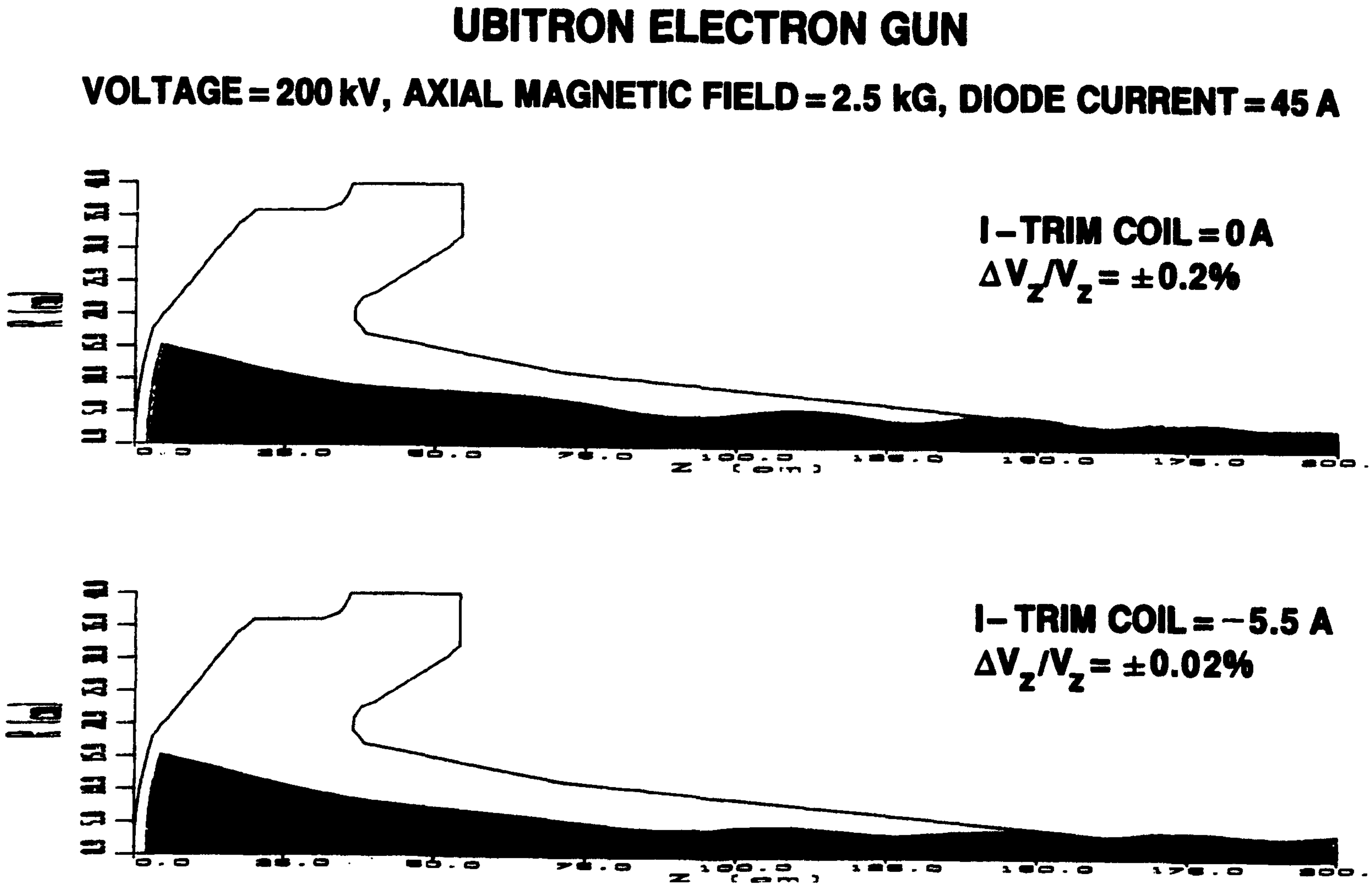


Figure 2.4. Laminar beam electron trajectories with different trim coil currents.

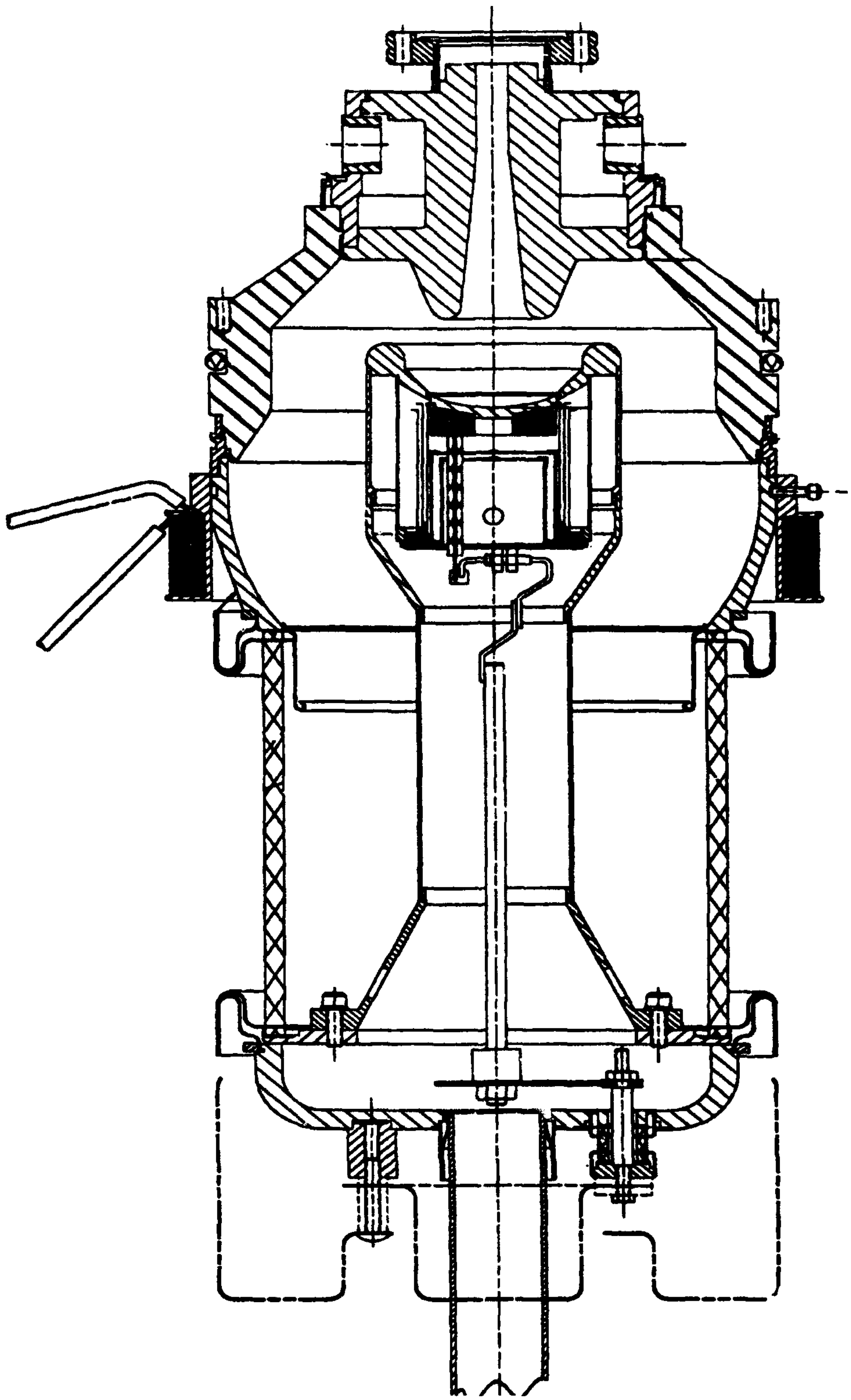


Figure 2.5. Schematic of advanced gun.

Before delivery to NRL, the gun was tested in a low voltage (<20 kV) beam analyzer. The measured electrostatic beam profile compared well with computer predictions concerning beam diameter, perveance, and beam minimum position. Confined flow test results were also good. Measured beam scalloping was less than 3% for $1.77B_{br} < B < 2.75B_{br}$, and without change in beam diameter and where the Brillouin field, $B_{br} \sim 880$ G. Computed velocity spread is less than 0.3%. A measured confined flow 2-D beam profile is shown in Fig 2.6.

ADVANCED GUN FOR THE NRL UBITRON

2 - D CURRENT DENSITY PROFILES

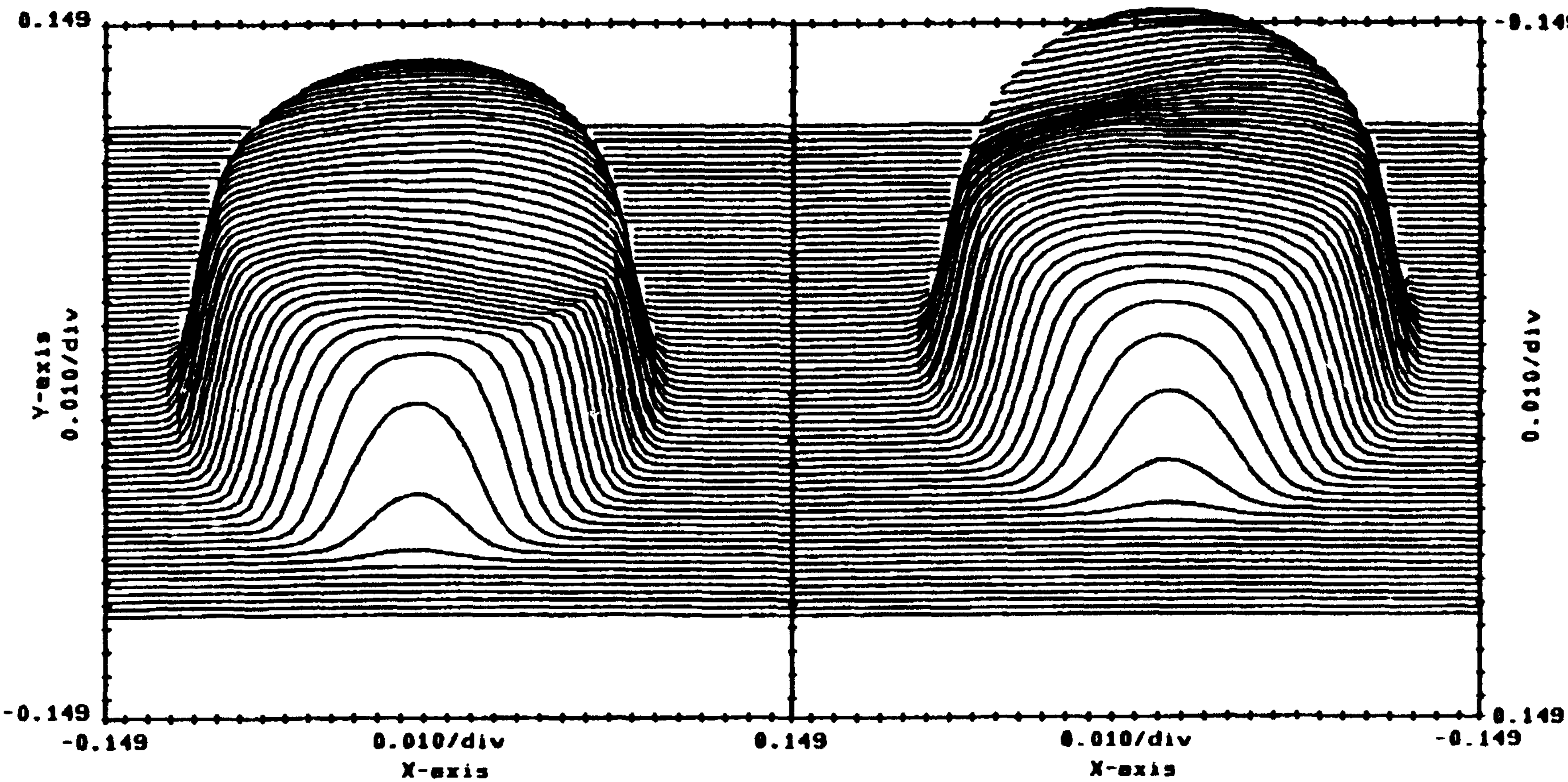


Figure 2.6. Measured confined flow 2-D beam profile.

2.2 CURRENT MONITORS AND BEAM COLLECTOR.

2.2.1 Current Monitors.

Due to the experimental nature of the ubitron, it is desirable to monitor the beam current at each of the major transition points of the tube - the electron gun region, and the entrance to and exit from the interaction region. The gun current is measured with a commercial current transformer, while the beam current at each of the other locations is measured using a resistive break in the return current path. The major design criteria for the beam current monitors were: field changeable resistance value, small gap to minimize RF perturbations, overall diameter less than the 2 3/4 inch Conflat flange diameter, and low self-inductance.

A photograph of the completed entrance current monitor is shown in Fig. 2.7. Each current monitor consists of four major components: 1) (two) waveguide/tube sections, 2) 5-mil Kapton film spacer, 3) resistor band, and 4) compression rings. The essential difference between the entrance and exit current monitors is the inner diameter, 4 and 8.15 mm, respectively. The compression rings insure good electrical contact between the fingerstock of the resistor band and the stainless steel tube body. Construction details are shown in Fig. 2.8.

Each resistor band consists of several non-inductive wire-wound resistors soldered between two fingerstock strips. Two resistance values are used, depending on the current emitted by the electron gun, 167 mΩ for the 37-A modified SLAC klystron gun, and 50-mΩ for the 100-A Varian gun. These values are chosen for a 5-V output signal at maximum current. The resistor band is demountable to prevent damage or resistance changes during tube bakeout. However, bakeout temperature is still limited by the Viton O-rings and Kapton spacer.

Electrical connection details and test set-up are shown in Fig. 2.9. The triaxial cable and 1:1 Mini-Circuits isolation transformer are used to minimize electrical noise and ground loop problems. Based on extensive testing with a fast rise time (<20ns), pulsed-current source, it was found that the best response was

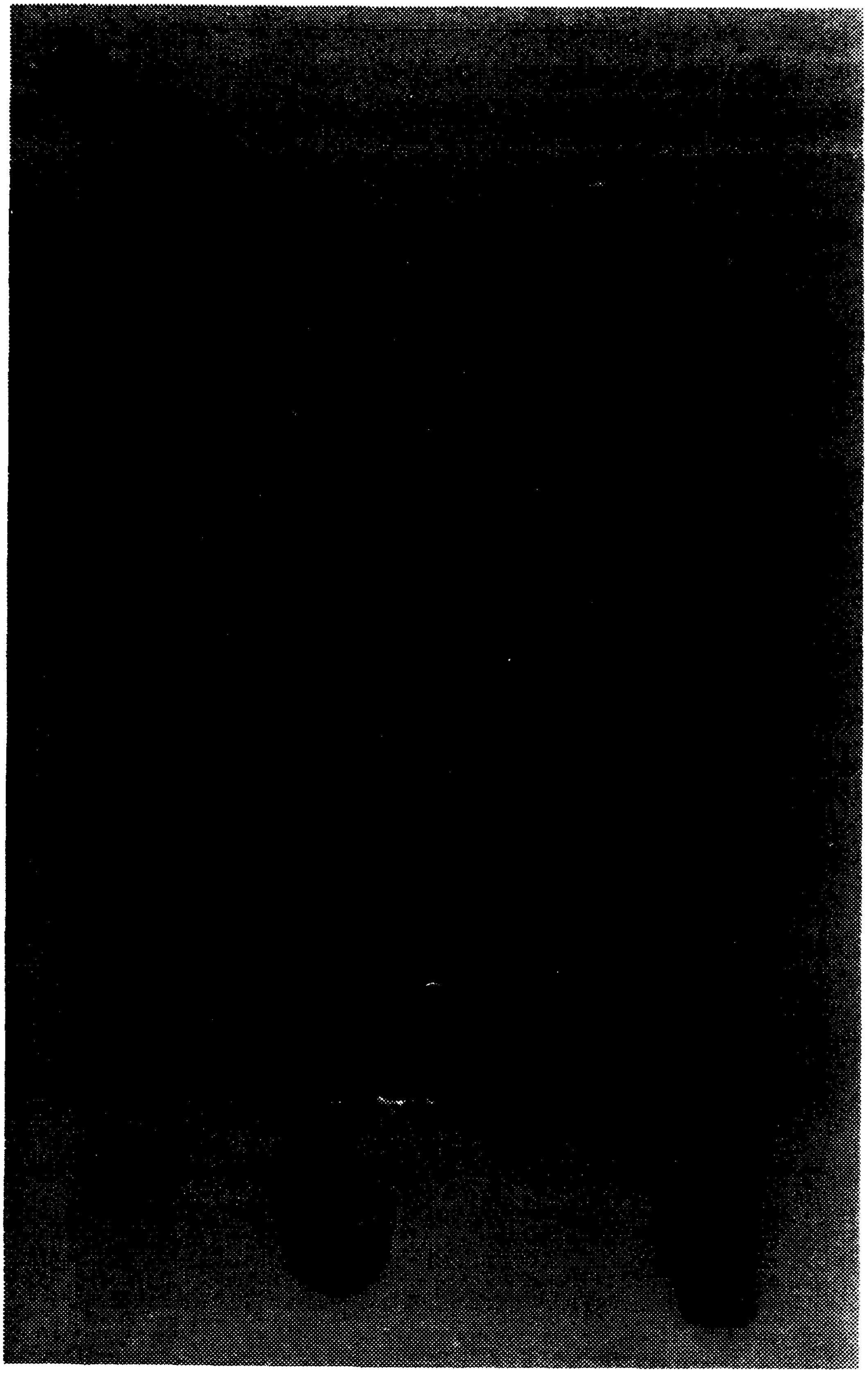


Figure 2.7.   Entrance current monitor.

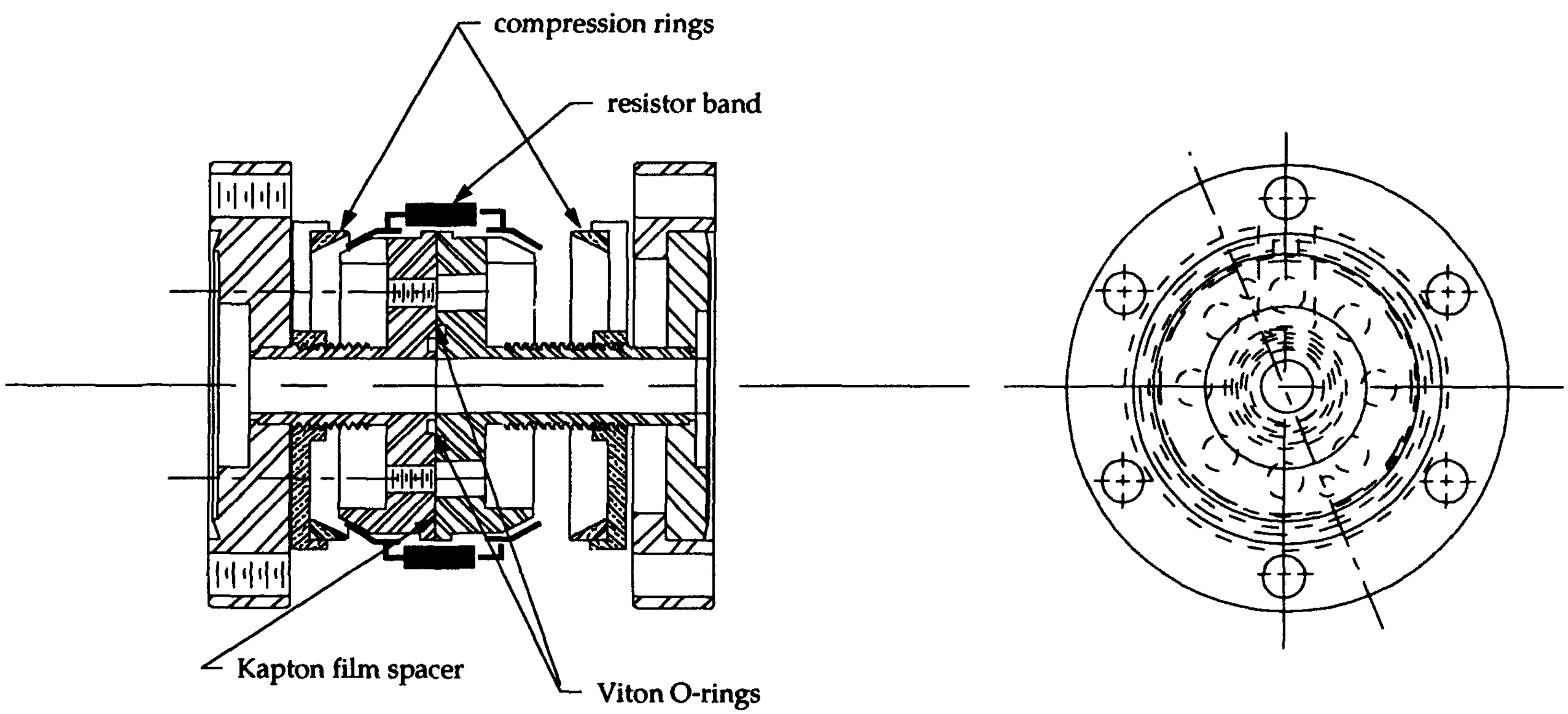


Figure 2.8. Current monitor construction details.

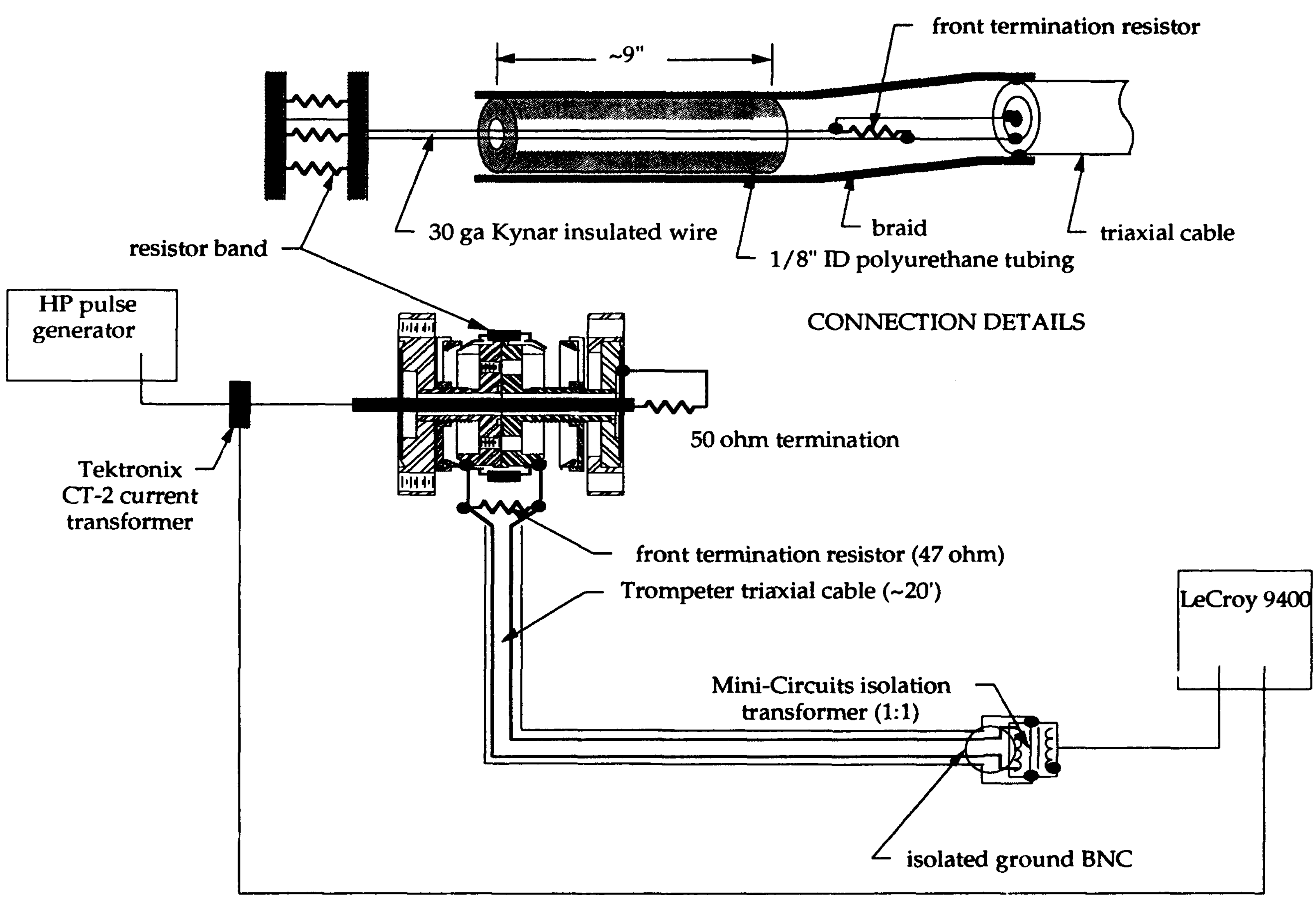


Figure 2.9. Current monitor electrical connections and test set-up.

obtained using the configuration shown. The resistor band connections were made with 30-ga. Kynar insulated wire inside a 9 in. long, 1/8 in. diameter polyurethane tube covered with the shielding braid from RG-223 coaxial cable. A 47-Ω source termination resistor is used for the best pulse response. An overlay of the input current pulse and the current monitor output signal is shown in Fig. 2.10. The resistance derived from these pulse tests, 164 mΩ, agrees quite well with the calculated and bridge measured value of 167 mΩ.

2.2.2 Beam Collector.

Following passage through the interaction region, beam electrons must be returned to the modulator. This is accomplished with a 'collector' or beam dump. By appropriately locating the collector in the fringing field of the solenoid, electrons following the diverging field lines will be directed into the collector walls. The beam, however, is still very powerful since only 15%, at best, of the injected beam power has been converted to microwave power.

To design the collector, we assume that the full beam power is absorbed. SCRIBE is used to calculate the current density distribution along the collector walls. With the collector entrance positioned at the solenoid pole piece, the peak current density is ~ 2.1 A/cm^2 for a 250-kV, 33-A beam, corresponding to a peak impulse wall loading of ~ 0.55 MW/cm^2. As a rule-of-thumb, this quantity should be less than 10 MW/cm^2 for a 1-μs beam pulse [12] to prevent localized heating on a timescale shorter than that for effective conduction cooling. Assuming a 1.5×10^{-4} duty factor, the highest average power density is ~ 80 W/cm^2, which is less than the 200 - 300 W/cm^2 rule-of-thumb. A diagram of the collector, showing water channels, is shown in Fig. 2.11.

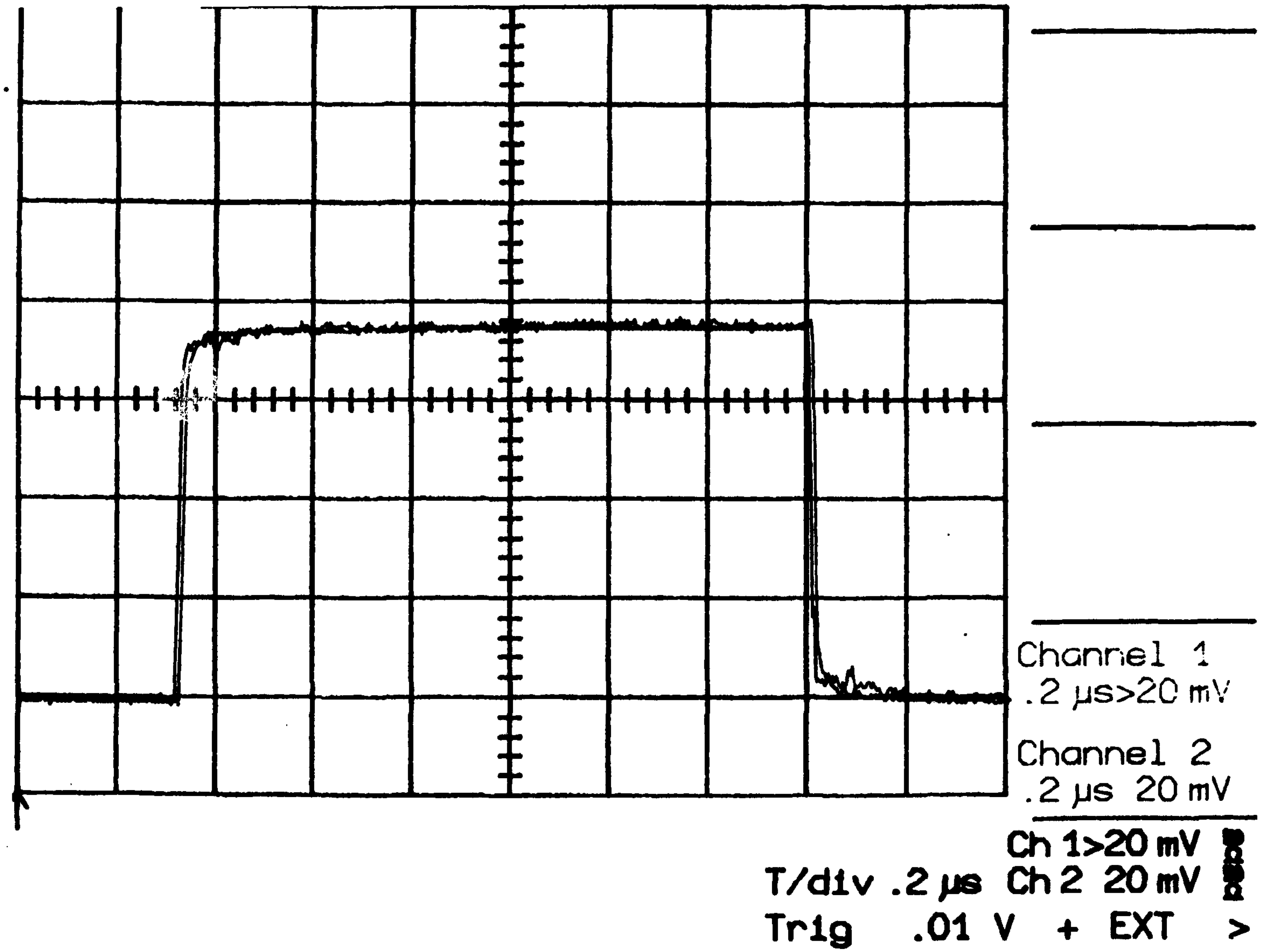


Figure 2.10. Current monitor test waveforms.

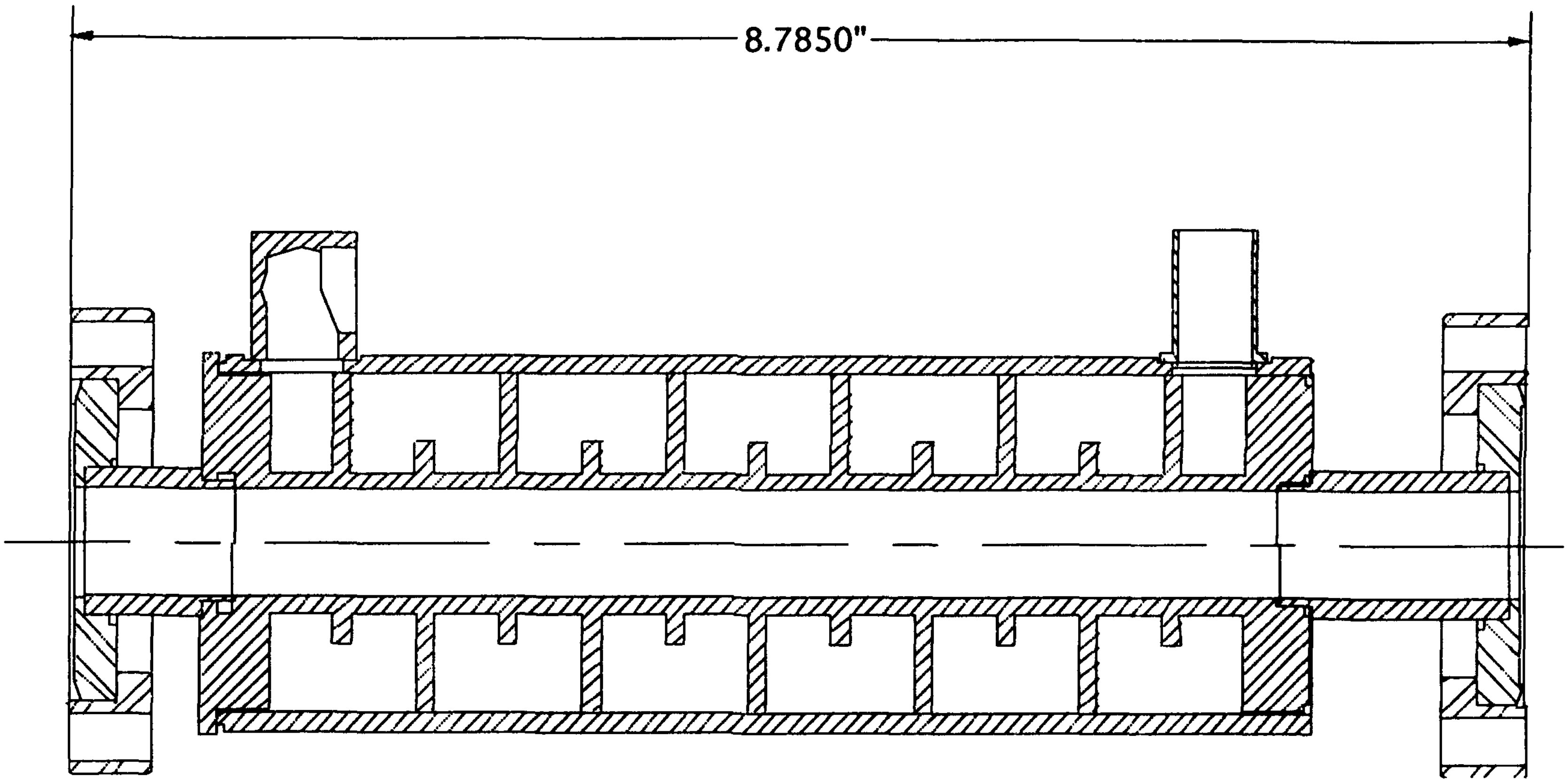


Figure 2.11. Beam collector.

2.3 SOLENOID.

An axial magnetic field is required to confine the beam for propagation through the ubitron, since wiggler field focusing alone is insufficient. While the inclusion of an axial magnetic field gives rise to a class of potentially competing instabilities (cyclotron maser), it also permits investigation of potential ubitron performance increases due to a gyroresonance of the wiggler and solenoidal fields.

The base configuration for the ubitron solenoid consists of fifteen 4-in. ID coils mounted inside a steel flux containment shell, including pole pieces. In order to generate a stable, well-controlled axial magnetic field, the water-cooled coils are driven with DC power supplies. An additional advantage of this configuration is the potentially wide variety of field profiles possible. The solenoid will generate an axial field of 3-kG, or better, at a drive current of 60 A. The maximum transverse field is specified to be less than 0.2% of the axial field, continuously, and is capable of generating a 5.5-kG axial field for short periods (~10 min.). This permits investigation of Group II orbit operation. A diagram of the assembled solenoid, overlaid with axial and transverse field measurements, is shown in Fig. 2.12.

In typical operation, the fifteen coils are divided into five three-coil groups, with each group driven in series by one power supply. For cooling purposes, a series-parallel water flow arrangement is used. Each three-coil group is series connected and fed in parallel from entrance and exit tap water manifolds. Flow meters are placed in each exit line to insure cooling flow is maintained at a flow rate in excess of the required 0.5 gpm (see Fig. 5.10). Fluid and power supply lines exit the solenoid through two 3 1/4 in. gaps in the steel shell running the length of the solenoid.

As discussed in Sections 2.1.1 and 3.2, the pole piece at the gun end of the solenoid is not appropriate for laminar electron flow with the modified SLAC klystron gun. To rectify this situation, the pole piece was altered by increasing the inner diameter to ~7.4 in. and repositioning the gun axially. The calculated modified and original field profiles are compared in Fig. 2.13. For this graph, the origin is inside the solenoid. The original profile was used for the rippled

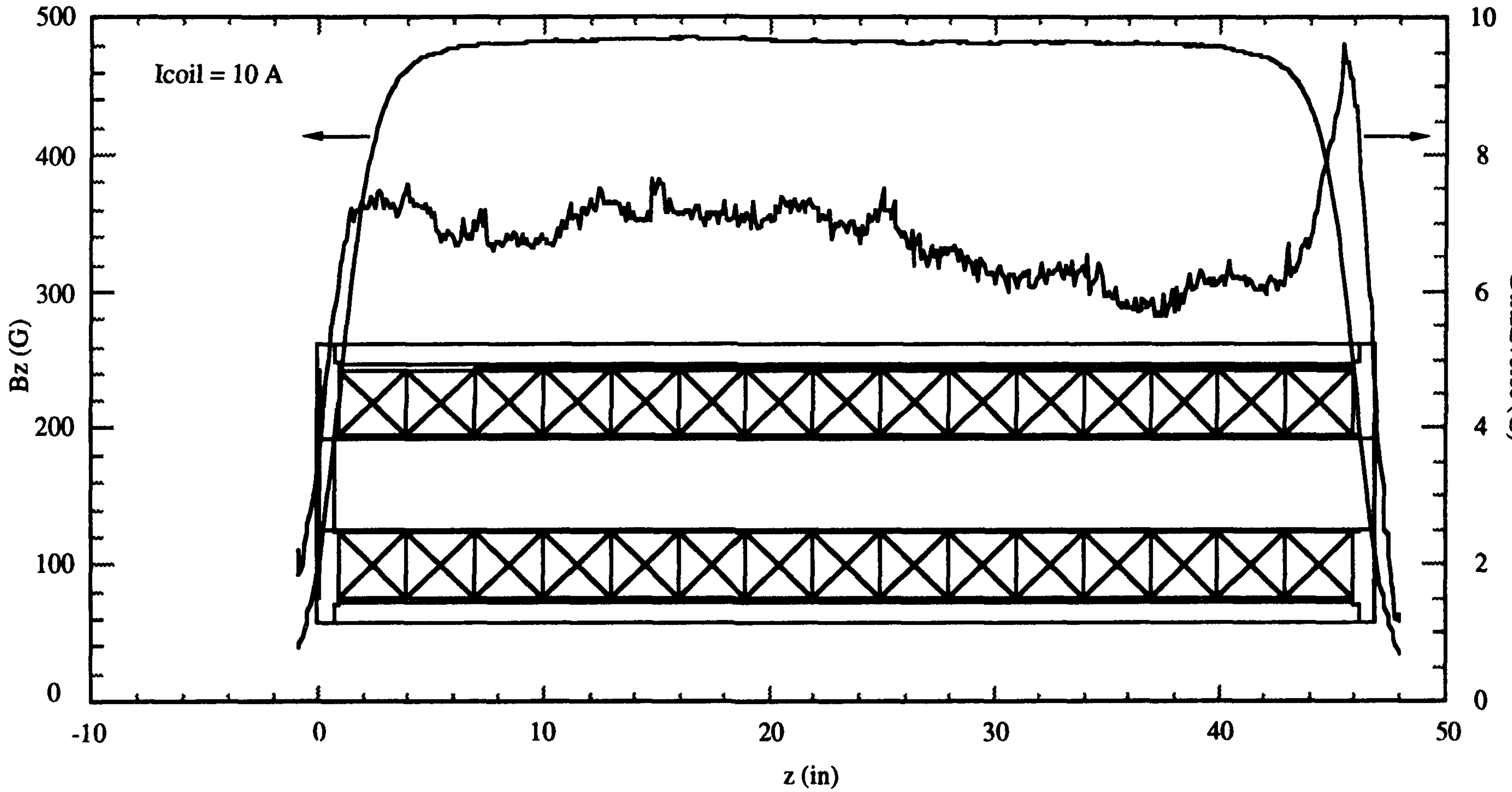


Figure 2.12. Solenoid axial and transverse field profiles.

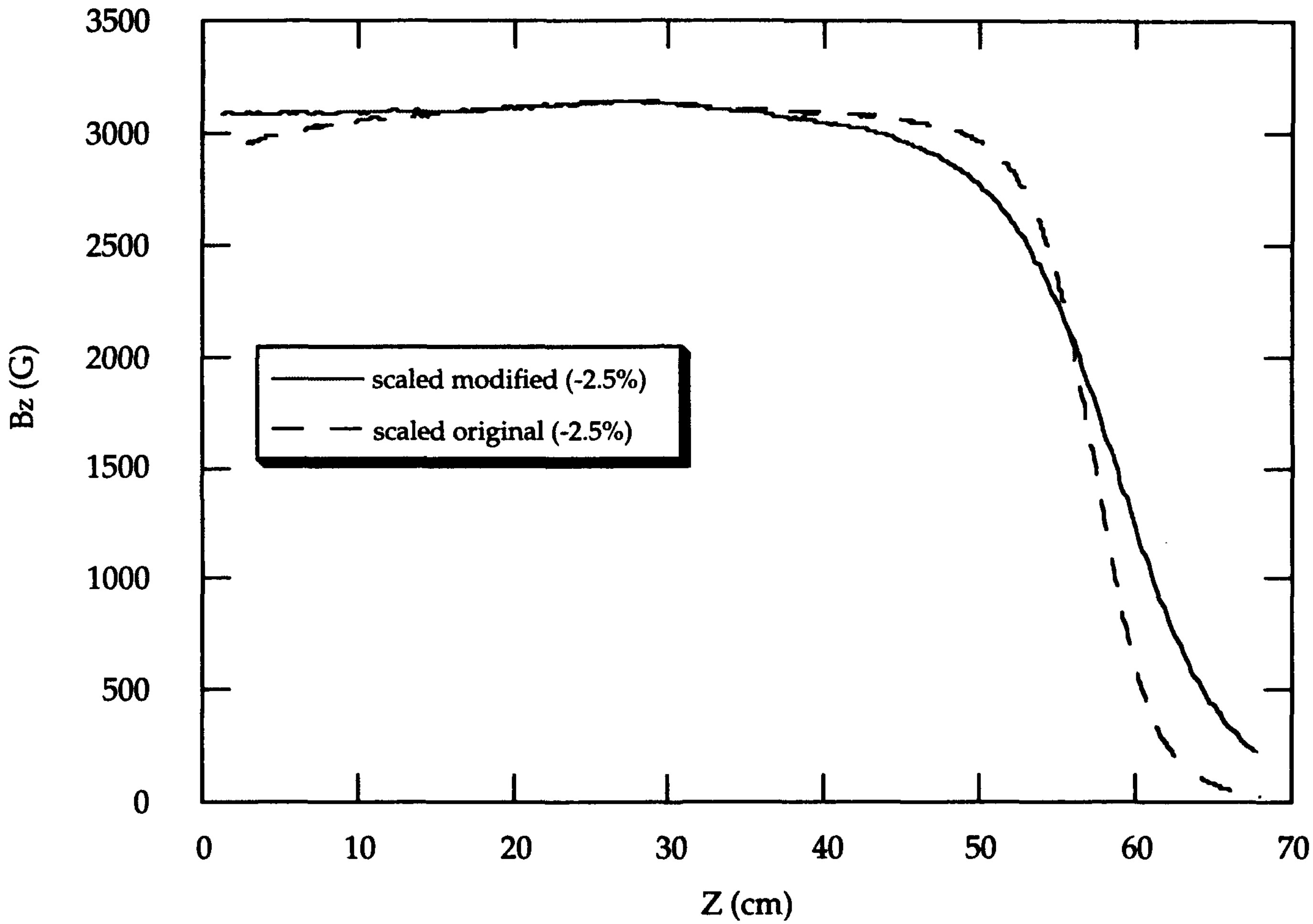


Figure 2.13. Axial field profile for original and modified pole pieces.

beam/harmonic ubitron experiments discussed in Section 3.1 and the modified profile was used in the fundamental mode ubitron experiments discussed in Section 3.2.

For use with the advanced gun at higher voltages, more substantial solenoid modifications were required in order to generate an axial field profile in the gun region that matched the 'goal curve' supplied by Varian. This profile insures maximum electron beam quality. With respect to the solenoid configuration, this gun differs from the SLAC gun in two major areas. First, the gun anode is made from Permandur iron and is part of the magnetic circuit, and must be inserted into the solenoid as opposed to attached to the solenoid. Second, the larger anode region diameter requires a larger coil ID than the original 4-in. ID coils.

With these differences under consideration, computer codes of the POISSON GROUP were employed to determine the optimum configuration for matching the goal curve. This involved a determination of the large coil dimensions and location as well as the optimum coil current distribution. The large coil outer diameter was constrained to fit within the inner diameter of the steel shell. The coil dimensions and location were first determined on a trial-and-error basis using POISSON. The results of the POISSON calculations were then used on an iterative basis with a Simplex optimization routine (Sec. 5.3) to calculate optimum coil currents for matching the goal curve.

A partial schematic of the modified solenoid configuration for use with the advanced gun is shown in Fig. 2.14. Magnetic materials are indicated by the shaded regions. With appropriate coil currents, this configuration results in a reasonable match to the goal curve, as shown in Fig. 2.15, where the coil labels are referenced to the previous figure. It should be noted that this is only the baseline profile; the ubitron has been operated with considerable departures from this profile.

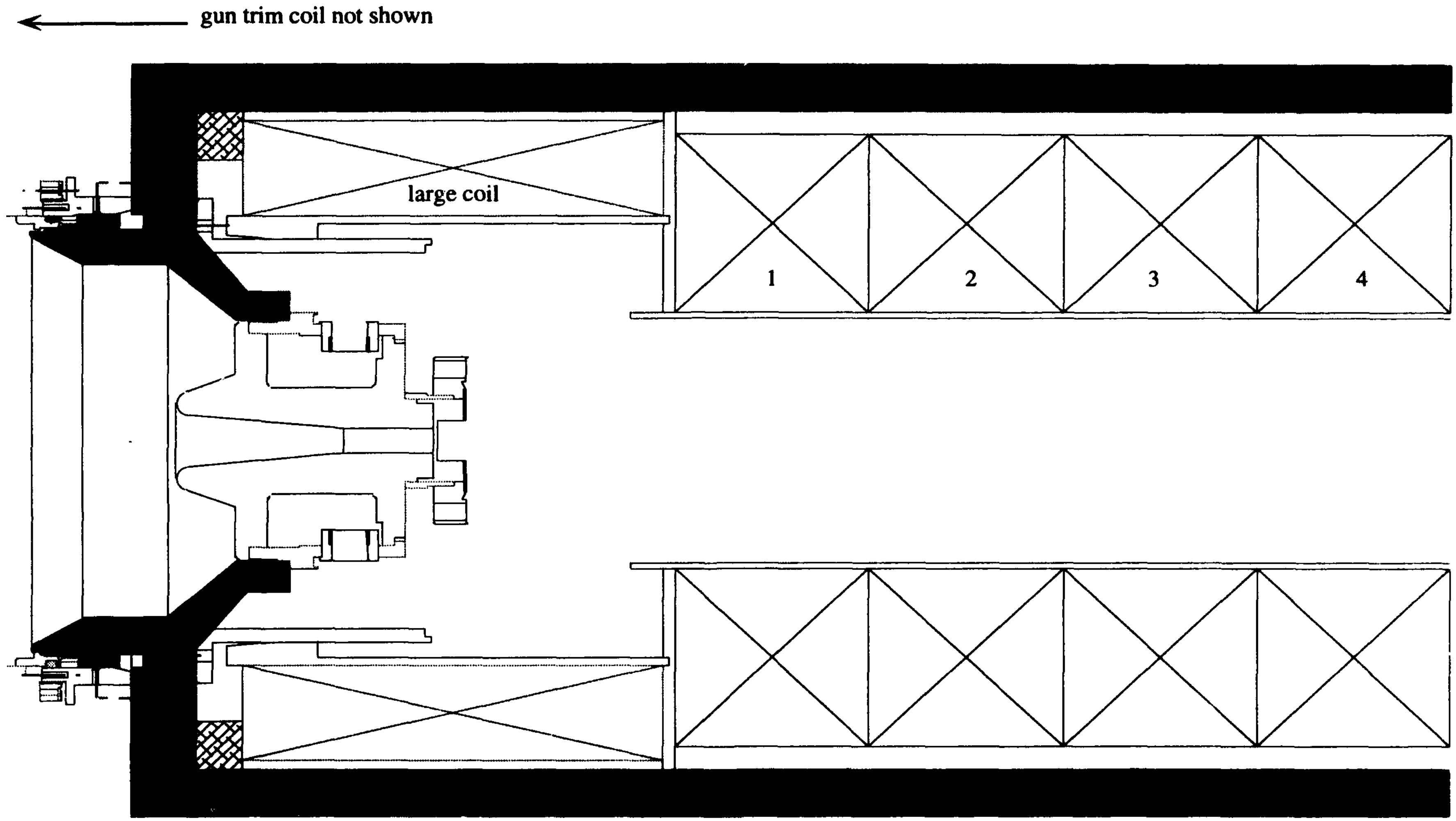


Figure 2.14. Modified solenoid configuration for advanced gun.

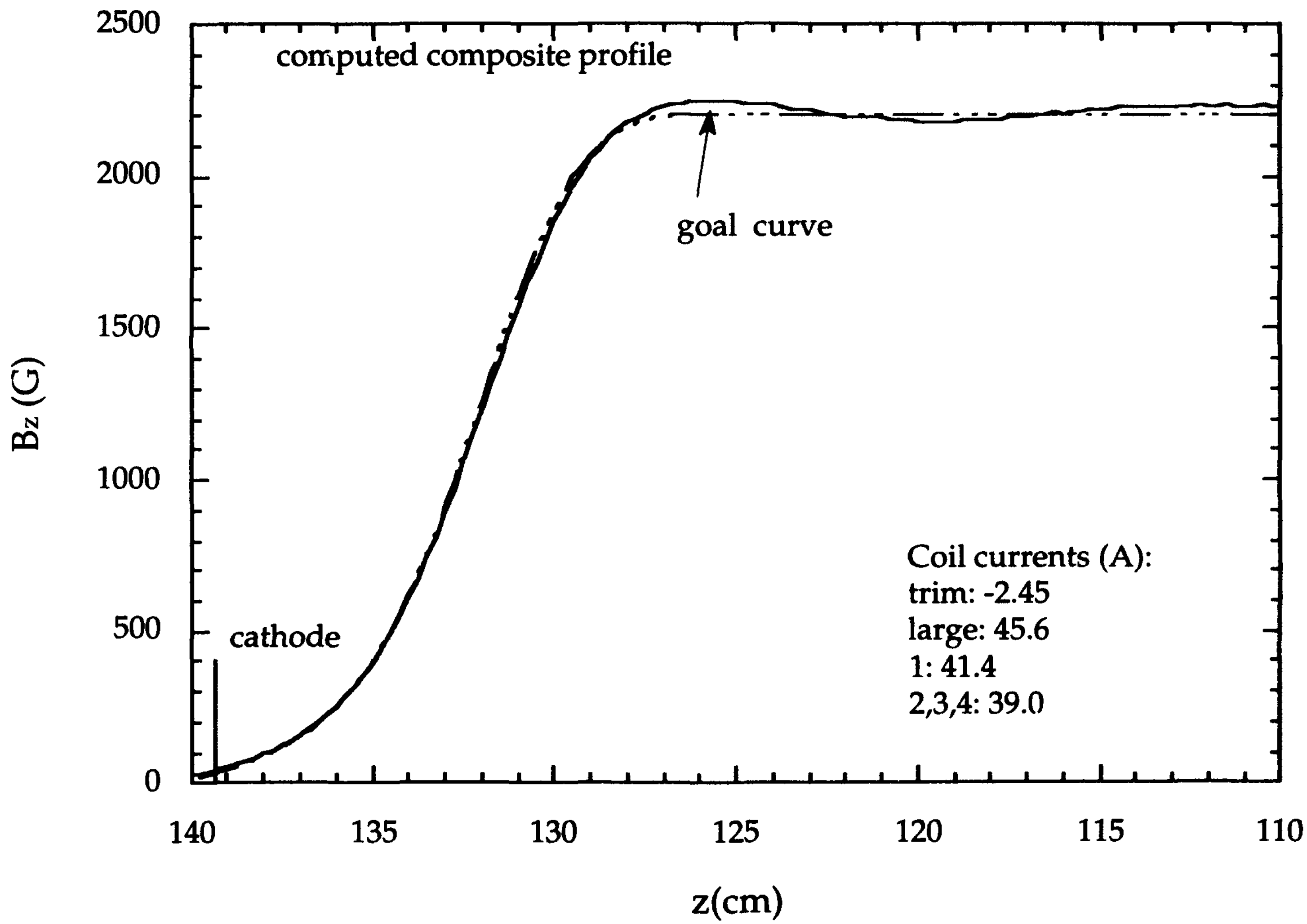


Figure 2.15. Axial field profile for advanced gun configuration, compared with goal curve.

2.4 WIGGLER.

2.4.1 Introduction.

The NRL ubitron is designed around a DC-operated, bifilar helix or wiggler. The wiggler must generate a right-hand, circularly-polarized, transverse field that adiabatically increases from zero to a chosen value, maintain this value for a sufficient number of periods in the interaction region, and then adiabatically decrease to zero again. The adiabatic field increase is an important requirement in order to access near ideal helical orbits that maintain beam quality and are readily modeled. DC operation is chosen to permit the attainment of accurate, repeatable, and continuously variable field values.

The maximum anticipated transverse field required is ~ 500 G, although typical operation is at fields of ~ 300 G. The accessible parameter space for the wiggler and axial magnetic fields is shown in Fig. 2.16. Since the transverse field generated by a bifilar helix is basically the remnant field of two solenoids with opposite currents, the generation efficiency, or gauss/amp, is relatively low. That is, relatively large drive currents are required to generate sufficient transverse fields. This is especially apparent with DC operation, where high currents translate to high power dissipation and possible wire insulation failure. The wiggler design is, therefore, based on a compromise between the conflicting requirements of high-field generation and low-power dissipation. A power dissipation of 10 kW is chosen as a reasonable limit for this design. Operation at even this value requires a cooling system. Irrespective of operational power limits, the wiggler must still be fabricated from materials compatible with high temperature bakeout procedures.

2.4.2 Design Procedure.

The basic wiggler design starts with a specification of the desired on-axis transverse field profile, with primary emphasis placed on the entrance ramp from zero to the interaction field value, B_0. The winding geometry, materials, and fabrication techniques that will generate the desired field profile, consistent with normal operating conditions, are then determined .

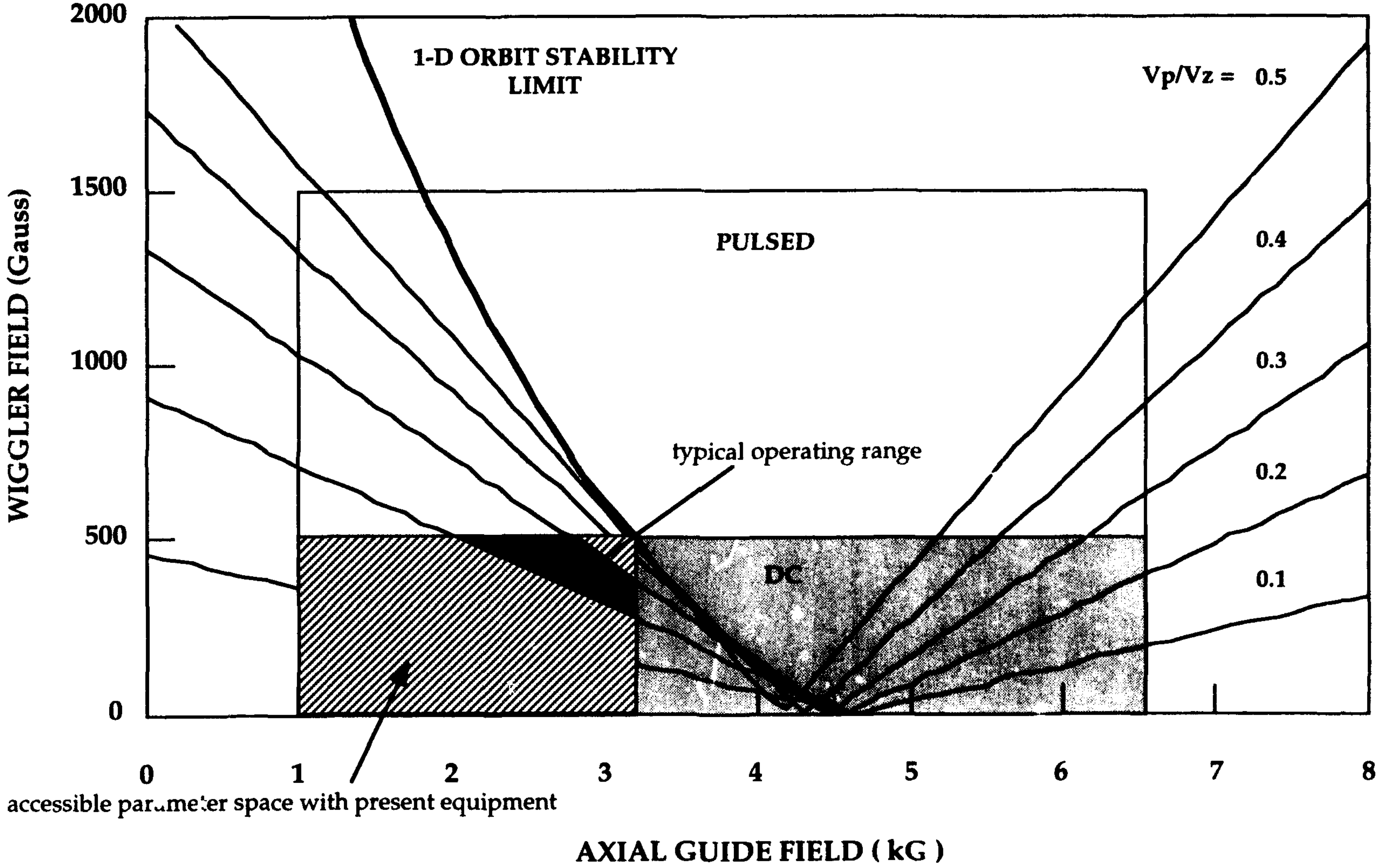


Figure 2.16. Accessible parameter space for axial and wiggler magnetic fields.

After investigation of several possibilities, the chosen entrance field profile is a cubic spline taper over five wiggler periods with zero derivatives at the starting and ending points. The exit profile is a similar down taper over three wiggler periods. A twelve-period uniform or interaction section should permit realization of the desired 25 - 30 dB gain.

The most straightforward method to generate the entrance and exit field tapers is to slowly flare the windings radially over the desired number of wiggler periods. The major drawback to this method is the generation of a field bump caused by the termination of helical symmetry and by the end loop necessary to connect the '+' and '-' helices. With a careful choice of radial taper profile, this field bump can be minimized while still generating the desired field taper.

A close approximation to the desired radial taper may be obtained from an inversion of the formula for the on-axis transverse field of an ideal, infinite single wire bifilar helix of radius a_0, period λ_w, and current I: $B_0 = 2k_wI/5[xK'_1(x)]$, where $x = 2\pi a_0/\lambda_w$, and K1 is a modified Bessel function. At any point, z, in the specified field taper, f(z), the winding radius, a, is calculated to be the radius that would generate the field $B_0f(z)$ using the above ideal formula. That is, the following equation is solved for x(z), and, hence, a(z):

$$x(z)K_1'(x(z)) = \left[x_0K_1'(x_0)\right]f(z), \text{ where } x_0 = \frac{2\pi a_0}{\lambda_w}, \text{ and } x(z) = \frac{2\pi a(z)}{\lambda_w}.$$

However, this formulation would require an infinite radius in order to generate zero field at the beginning of the taper. To eliminate this problem, a maximum winding radius is first chosen, consistent with space constraints. Rather than abruptly terminating the radial flare at this point, the taper is actually terminated earlier, with a conical extension then added between the termination point and the maximum winding radius chosen above. The termination point is chosen such that the tangent to the computed flare profile intersects the z=0 axis at the maximum winding radius. The computation is carried out by the code INVHLX listed in Section 5C.

While the above calculation ignores the end-loop contribution to the actual field taper, it is included in an analytic Biot-Savart calculation of the field profile using the complete winding profile determined above. Fig. 2.17 shows the

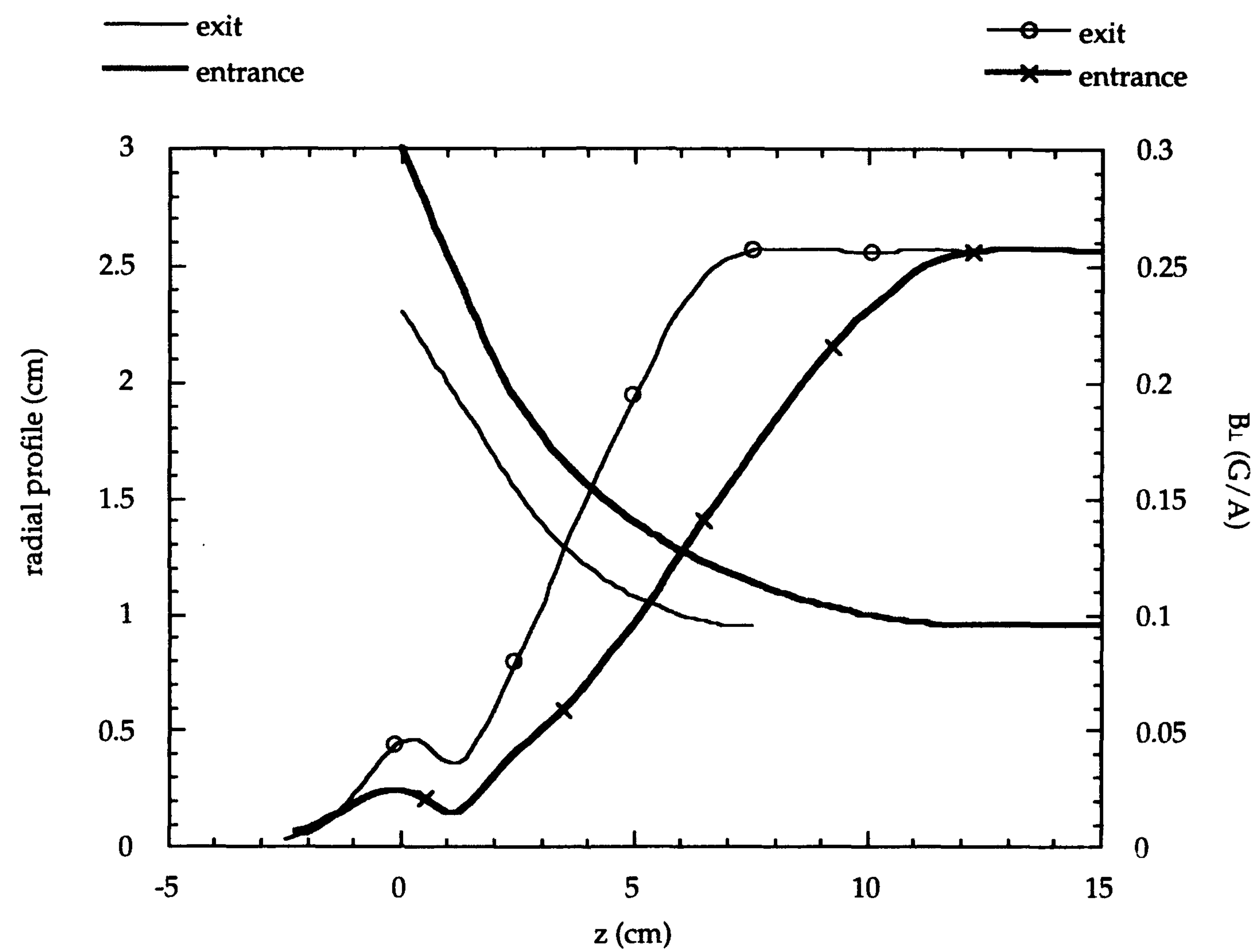


Figure 2.17. Wiggler taper profiles and calculated field profiles.

winding profile calculated to generate a cubic spline field profile and the resulting Biot-Savart field calculation for both entrance and exit tapers. Examining the entrance taper, the field is seen to increase smoothly with only a ~ 10% bump. Due to the smaller maximum winding radius and shorter taper length, the bump at the exit taper is somewhat larger.

2.4.3 Electro-Mechanical Configuration.

Physical realization of the ideal profile described above is somewhat more involved. Approximately 2 kA would be required to generate a 500-G transverse field for a single-wire, bifilar helix built to the above profile. This is obviously excessive for a reasonable DC power supply. The most straightforward solution is to increase the number of windings or turns per helix. Wire dimensions must be small enough to maximize the number of turns in the available space, while also being large enough to minimize the total winding resistance to match to power supplies and minimize power dissipation. The final winding configuration consists of 10 turns of 32 x 90 mil Polythermaleze insulated magnet wire per helix, edge wound in a ~ 0.350 in. wide winding channel. With series-connected windings, the calculated resistance is approximately 350 mΩ. While axially spreading the conductors by approximately 1/3 of a period reduces the generation efficiency by ~ 25 % to ~ 1.85 G/A, 500-G is now attainable with ~ 275 A.

To support the conductors in the specified winding profile, a winding form was machined, using a numerically controlled lathe, from a solid block of aluminum and then epoxied to the stainless steel waveguide. To obtain the strongest uniform field possible with a given current, the form is machined down to the supporting waveguide in the interaction section. This will place the conductors as close as possible to the waveguide axis. Details of the winding form are shown in Fig. 2.18.

The final design issue is the configuration of the end pieces, required to connect the windings of the '+' and '-' helices. To match existing DC power supplies, the windings are series-connected. This increases the required power supply voltage while reducing the current. Unlike a single-wire, bifilar helix, this necessitates end loops at both ends of the wiggler. As described below, the final end piece configuration succeeded several unsuccessful versions. An

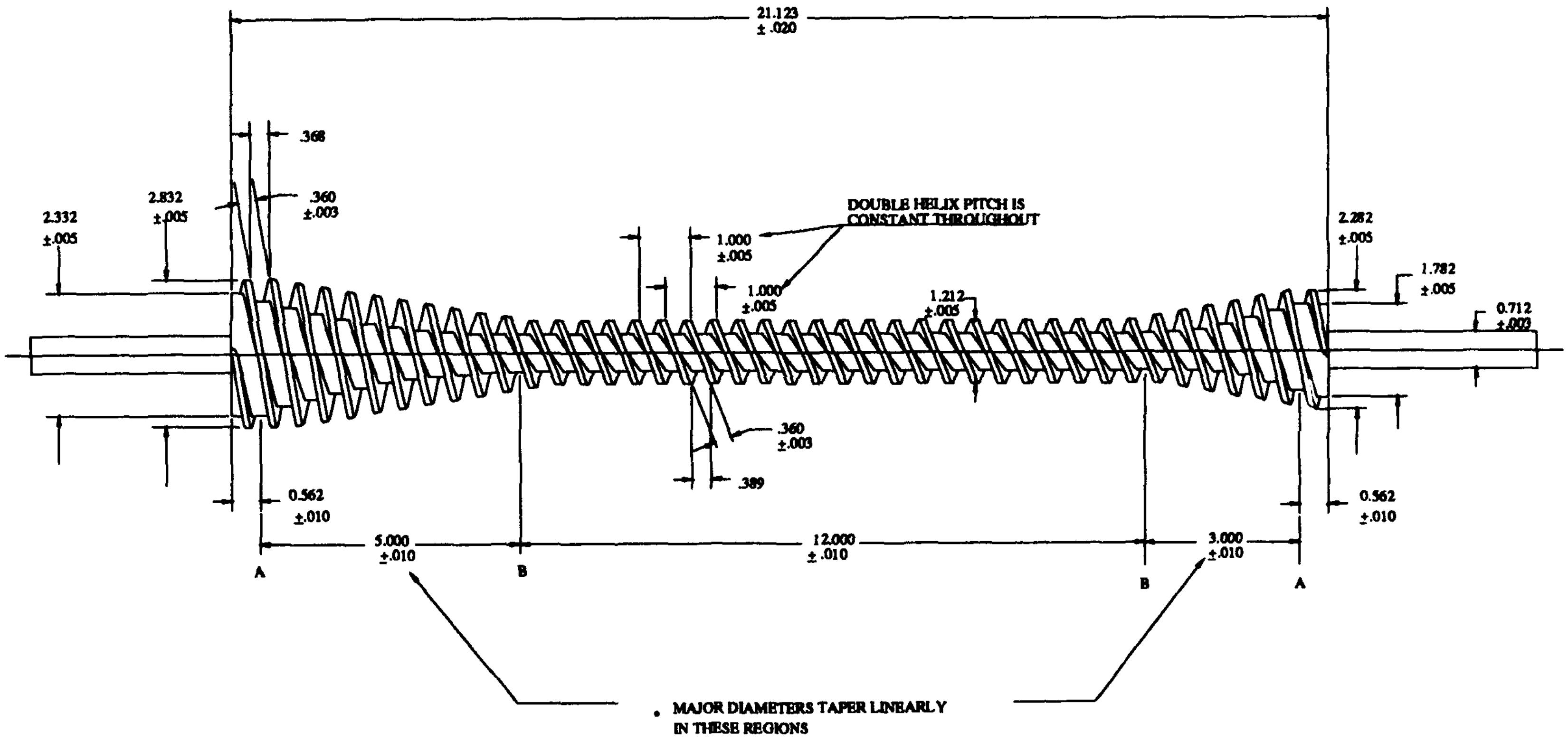


Figure 2.18. Wiggler winding form.

engineering drawing of this configuration is shown in Fig. 2.19. While somewhat difficult to describe, its basic function is to separate and hold the end loop strips to permit coolant flow between the individual strips and down the length of the helix. A simplified schematic of the end loop connection scheme for a four-wire/helix configuration is also shown (Fig. 2.20.). The current path is as follows: input is connection [5], current flows down the '+' helix to connection [v] at the gun end, then around the end loop to connection [i], then returning to the calorimeter end through the '-' helix to connection [1], around the end loop to connection [6], to [vi], to [ii], to [2], ..., to the exit connection [8].

For DC operation, ~ 10 kW (50 V, 200 A) of power can be dissipated in the wiggler, necessitating a cooling system. The major components of this cooling system are depicted in Fig. 5.10. It should be noted that the actual coolant fluid is Fluorinert FC-77 manufactured by 3M, and is chosen for the following properties: chemical inertness, high voltage breakdown strength, and heat transfer capability. A chilled water heat exchanger cools the heated fluid which is recirculated through the wiggler with a high pressure pump. A solid aluminum insert (2 pieces) is added between the wiggler winding form and fluid jacket in order to force coolant flow over the windings. If cooled before ubitron operation, this insert also acts as a heat sink.

2.4.4 Wiggler Assembly and Tests.

Several assembly difficulties led to modifications of the design presented above. The primary problem was shorting of the outermost windings to the aluminum form, caused by abrasion of the wire insulation during winding. Also, the total measured wiggler resistance, including end loops and lead wires, turned out to be considerably higher than calculated. This limited the maximum attainable field to an unacceptably low value. Additionally, large transverse field bumps (~30-50% of B_0) were measured using axial stacking of the end loops on a G-10 form, rather than the radial stacking configuration described above.

It was determined that use of the central eight wires of the ten wire bundle as conductors, with the outer two wires electrically disconnected and used as a mechanical buffer, was an acceptable solution to the resistance and shorting problems. This reduced the generation efficiency to 1.56 G/A, corresponding to a

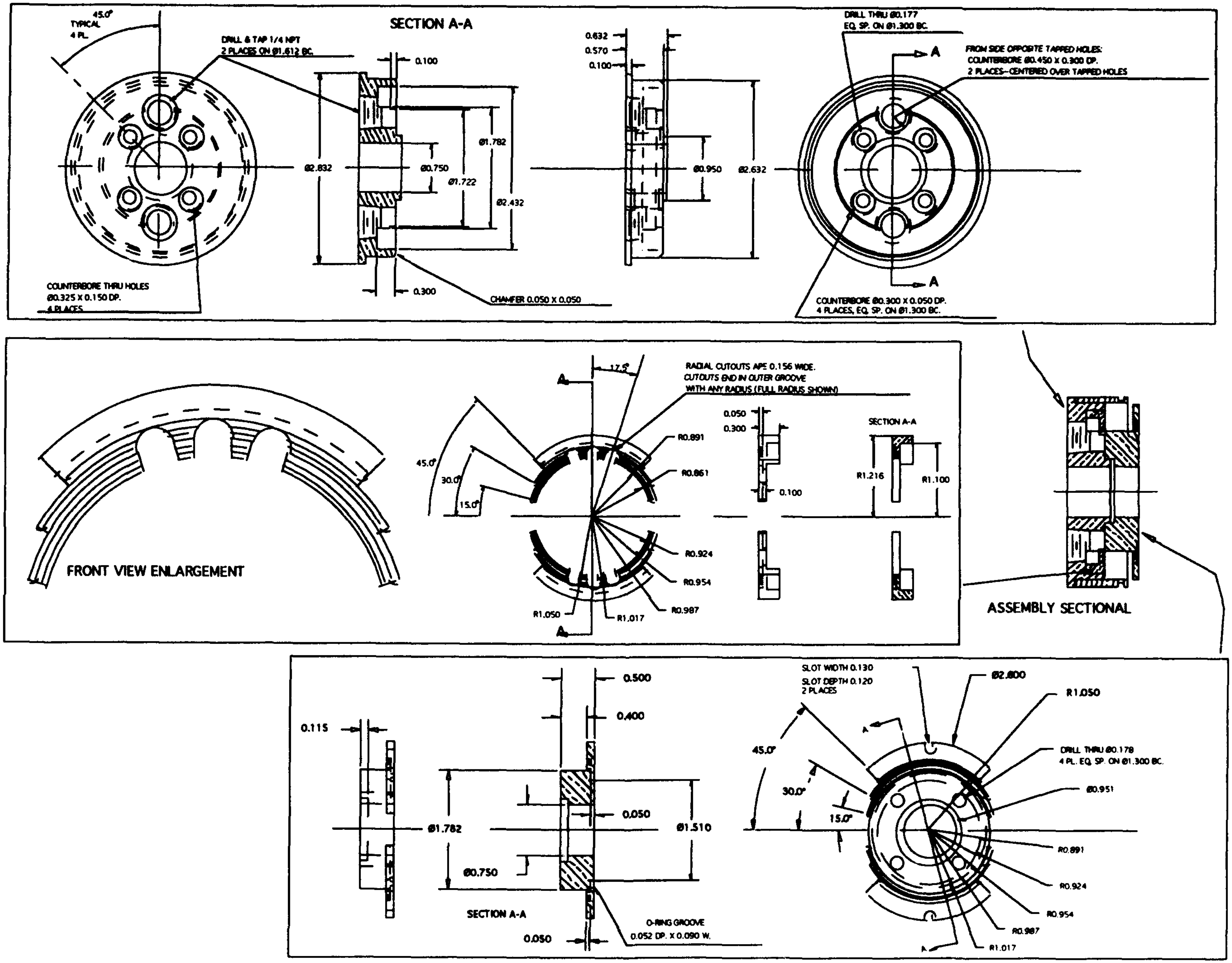


Figure 2.19. Wiggler end piece configuration.

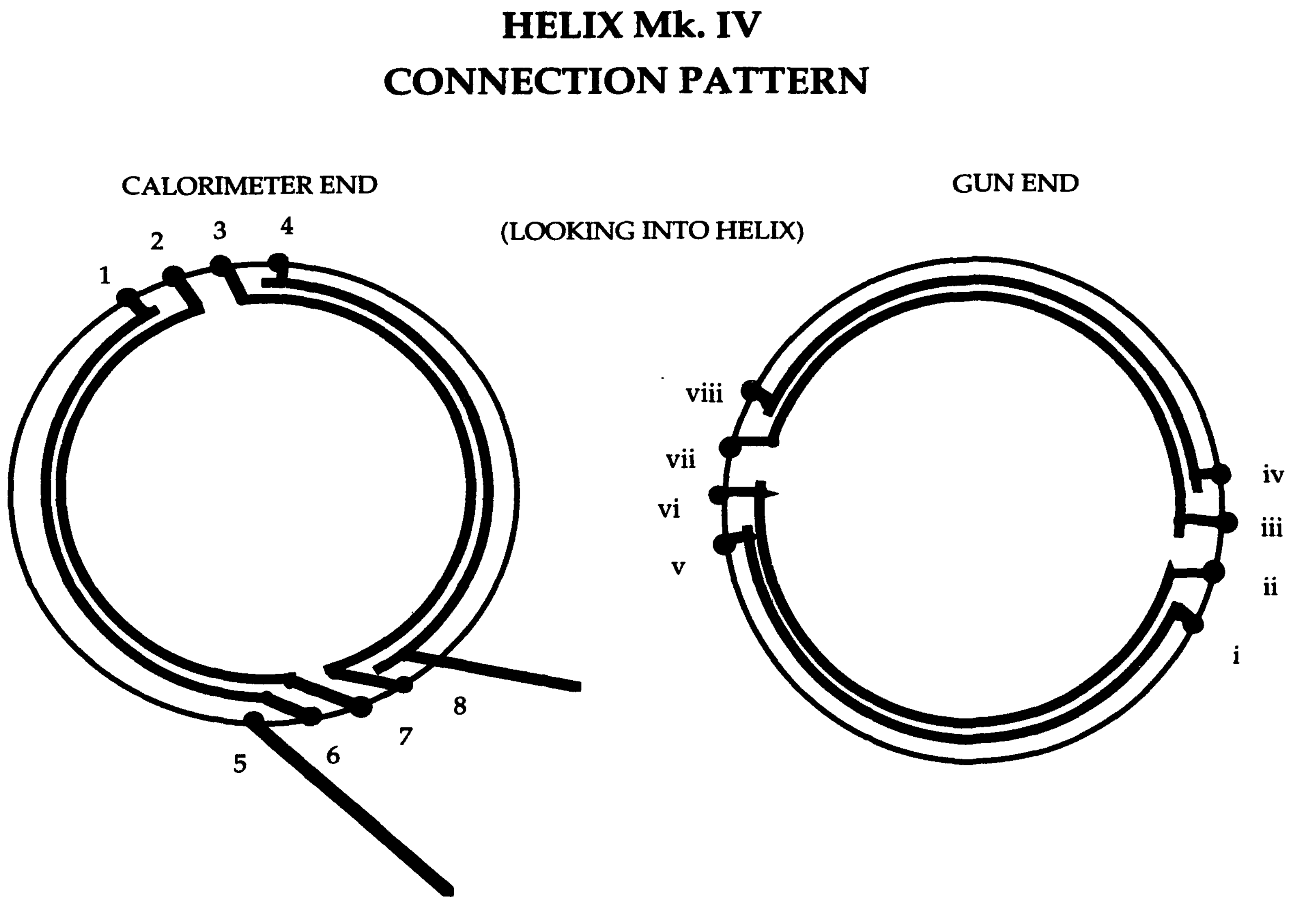


Figure 2.20. Four-wire end piece connection scheme.

265-G maximum field using a 50-V, 200-A power supply and the calculated wiggler resistance.

The transverse field of this 8-wire/helix configuration is shown in Fig. 2.21, measured at 50-A with an F. W. Bell BH-703 3-axis Hall probe. Ripples of substantial amplitude are observed in the 'uniform' section of the wiggler. However, this is not a true measurement of the on-axis transverse field since the B_X probe element is displaced from the axis by approximately 1.7 mm. To ascertain the contribution of the probe displacement to the observed ripples, a computer code, RIBTST2 (Section 5.C), was written to calculate the field normal to each probe element, averaged over the finite element dimensions, with each element location and orientation taken into account. The transverse field based on this calculation is also shown in Fig. 2.21, shifted axially to align with the initial ripples. The agreement between measurement and calculation is seen to be quite good, indicating that the actual on-axis transverse field is very close to that produced by an ideal bifilar helix. A second feature noted in the measurement is a downward step in field for the last two periods of the uniform section and the exit ramp. This was caused by an error in instructions for the numerically controlled lathe that shifted the inner radius of the winding form outward by approximately 1.1 mm.

The configuration described above worked quite well during initial ubitron operation. However, a loss of coolant during operation resulted in severe damage to the wire insulation, shorting the windings together in many places. In order to maximize time for ubitron operation, wiggler repair was attempted without tube disassembly, keeping the tube under vacuum. The same winding configuration could not, however, be maintained, since the tension required to edge-wind the rectangular wire would have buckled the unsupported tube.

An acceptable replacement winding geometry was found to be four round 12-ga. magnet wires, 80-mil diameter, 2-mil Polythermaleze insulation. These wires are not as precisely located as originally, since the combined wire width is slightly less than the channel width. Also, it was difficult to maintain a tight wire pack due to wire stiffness. The most serious drawback to this configuration, however, is the substantial reduction in generation efficiency to ~ .72 G/A. The

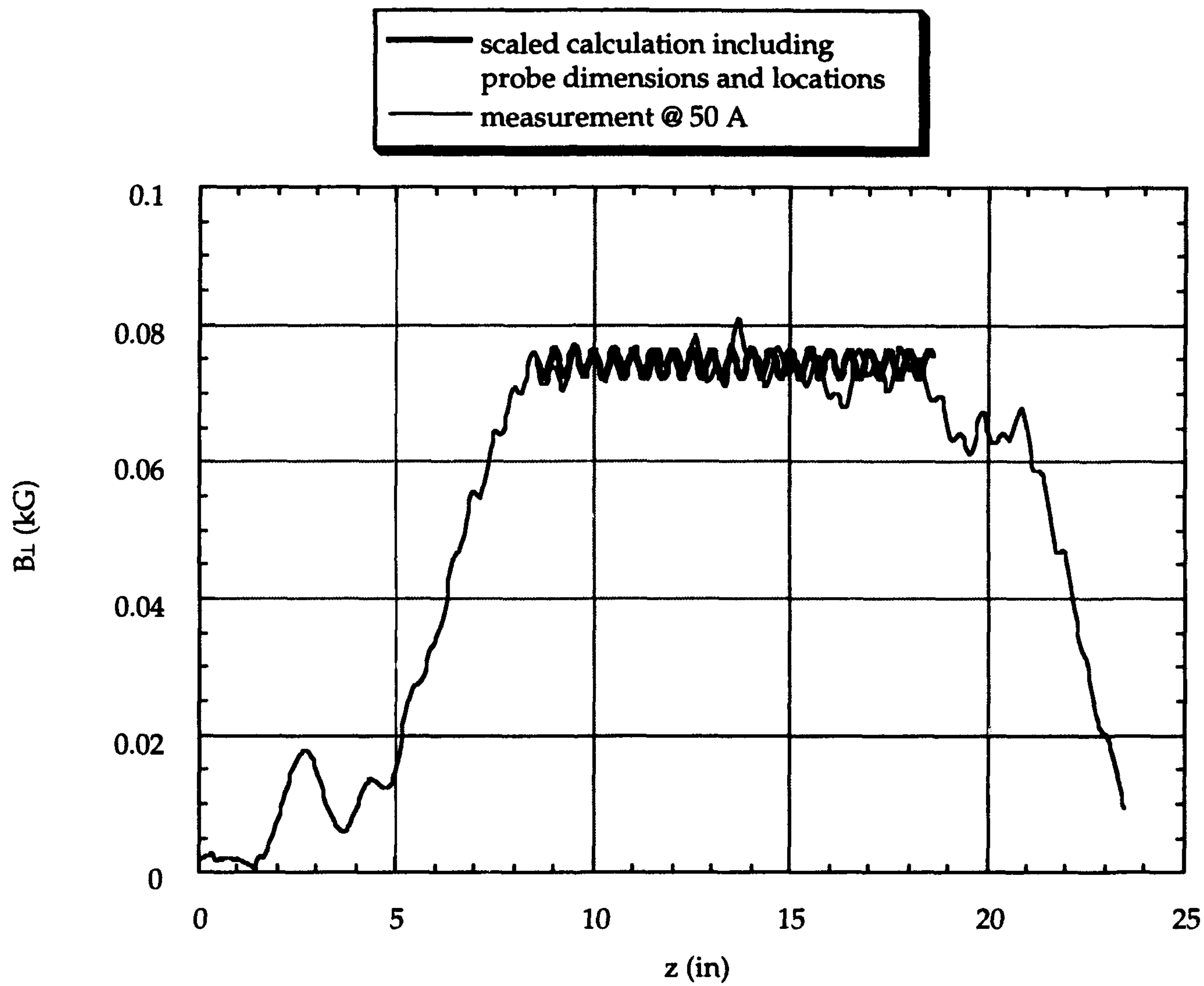


Figure 2.21. Measured transverse field of eight-wire wiggler; calculated field averaged over probe area.

maximum transverse field attainable is only 140 G, at the measured resistance of 98 mΩ and 200 A. This necessitated a shift from DC to pulsed operation in order to attain sufficient drive current to reach 500 G.

Although finding a suitable pulser was difficult due to the unusual parameter regime (low voltage, high current), a repetitive pulse variant of the single-shot 'Slapshot' pulser, manufactured by Plasma Research Corporation, was obtained. This is an SCR-switched capacitor pulser that nominally delivers 200 V pulses into loads up to 0.1 Ω at repetition rates up to 30 Hz. The current pulse into the wiggler load is essentially constant for ~ 10 μs, which is much longer than the beam pulse.

Unfortunately, measurements of the revised wiggler field could only be made after the ubitron was disassembled, following extensive ubitron operation. Both DC and pulsed on-axis field measurements are shown in Fig. 2.22, overlaid onto a schematic of the winding form. A moderate field reduction in the uniform section with the pulsed mode is readily apparent and is especially severe for the entrance and exit ramps. This is caused by magnetic field diffusion through the aluminum winding form. Also note the severe reduction in field for the last two periods of the uniform section, caused by the mistaken radial step in the winding form discussed above. As in the earlier measurements, considerable structure is apparent in the 'uniform' section. A major contributor to this structure could be the winding form, since some of the features are common between the original and the revised wiggler, as shown in the overlay of B_y measurements for the 10 rectangular wire and 4 round wire versions (Fig. 2.23).

In general, the field of the revised wiggler with pulsed operation departs considerably from the desired profile, especially in the loss of adiabaticity of the entrance ramp. One beneficial aspect, however, is the reduction of the entrance field bump amplitude in comparison with the average field in the uniform section. Nevertheless, comparison of ubitron experimental results with theory is complicated by the difficulty in modeling the measured field for simulation purposes. This will be discussed later.

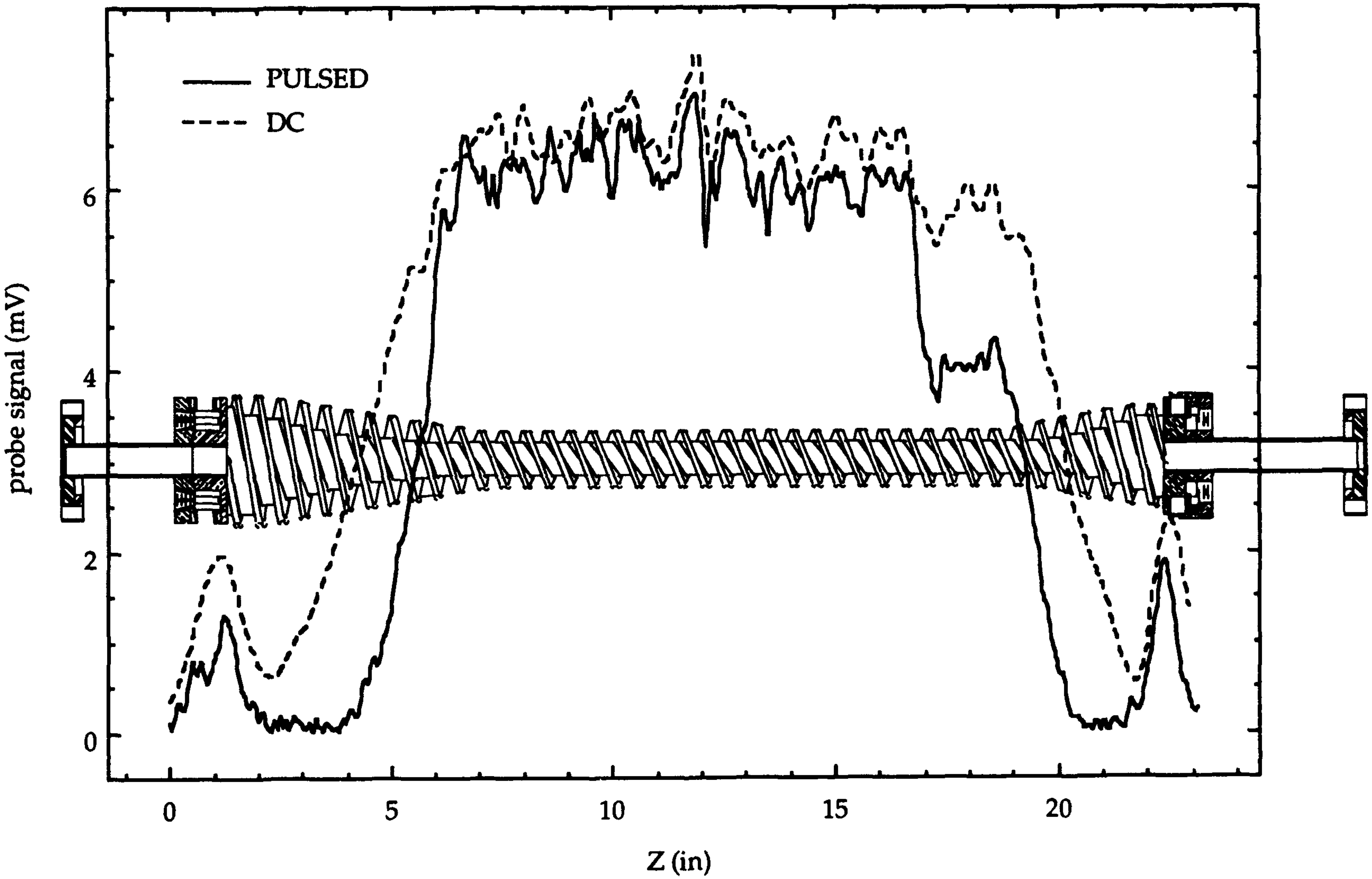


Figure 2.22. Measured transverse field of repaired wiggler, DC and pulsed.

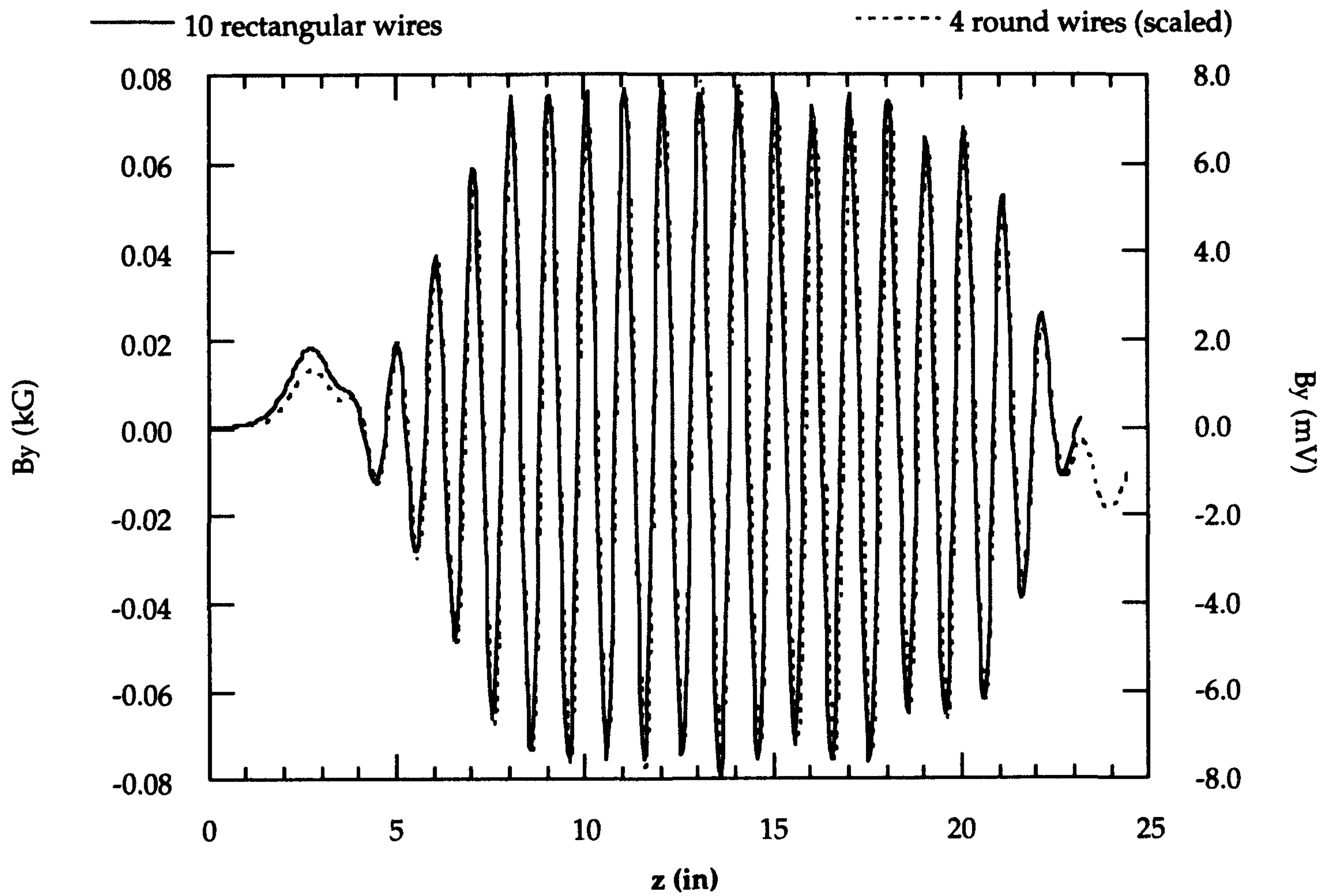


Figure 2.23. By for "original" and repaired wiggler.

2.5 INPUT COUPLER.

2.5.1 Introduction.

The function of the input coupler is to transfer microwave power from the driver through the vacuum envelope and inject a circularly polarized TE_{11} wave into the ubitron interaction region for amplification. The desired performance characteristics are as follows: broad instantaneous bandwidth > ±10% (13.5 - 16 GHz), with high power capability, ~ 10 kW. Since the ubitron/FEL mechanism amplifies an LHCP wave, the coupler must launch an LHCP wave in the direction of the load/calorimeter, only with less than 3 dB transmission loss. If a unidirectional, linearly-polarized wave were launched, power would be equally split between the LHCP and RHCP waves resulting in an effective 3-dB power loss. Additionally, the coupler must accommodate the electron beam and be of short axial length in order to minimize use of the solenoidal magnetic field.

2.5.2 Microwave Design.

A variety of coupler designs were considered, but most had significant deficiencies. For example, one design consists of two 3-dB directional couplers mounted orthogonally on square waveguide, which is then adiabatically deformed into circular waveguide. This is a wide band configuration that satisfies the unidirectional requirement. However, the effective transmission loss would be at least 3 dB, and the device would be excessively long.

The selected configuration, that meets all of the design criteria, is a modification of the 'turnstile' junction coupler, schematically shown in Fig. 2.24 [13]. A circularly polarized wave can be generated from the superposition of two linearly polarized waves with a 90° phase difference. For the standard turnstile coupler, one polarization is set by power injected into Arm 1 with Arm 2 terminated. Arms 3 and 4 are shorted at 5/8 and 7/8 guide wavelengths at the driven frequency, which creates 90° and 270° phase shifts at the respective arms, thus generating a linearly polarized wave orthogonal to the first with the necessary phase shift. Ellipticity increases as the operating frequency changes. As shown in the figure, several matching posts are typically required.

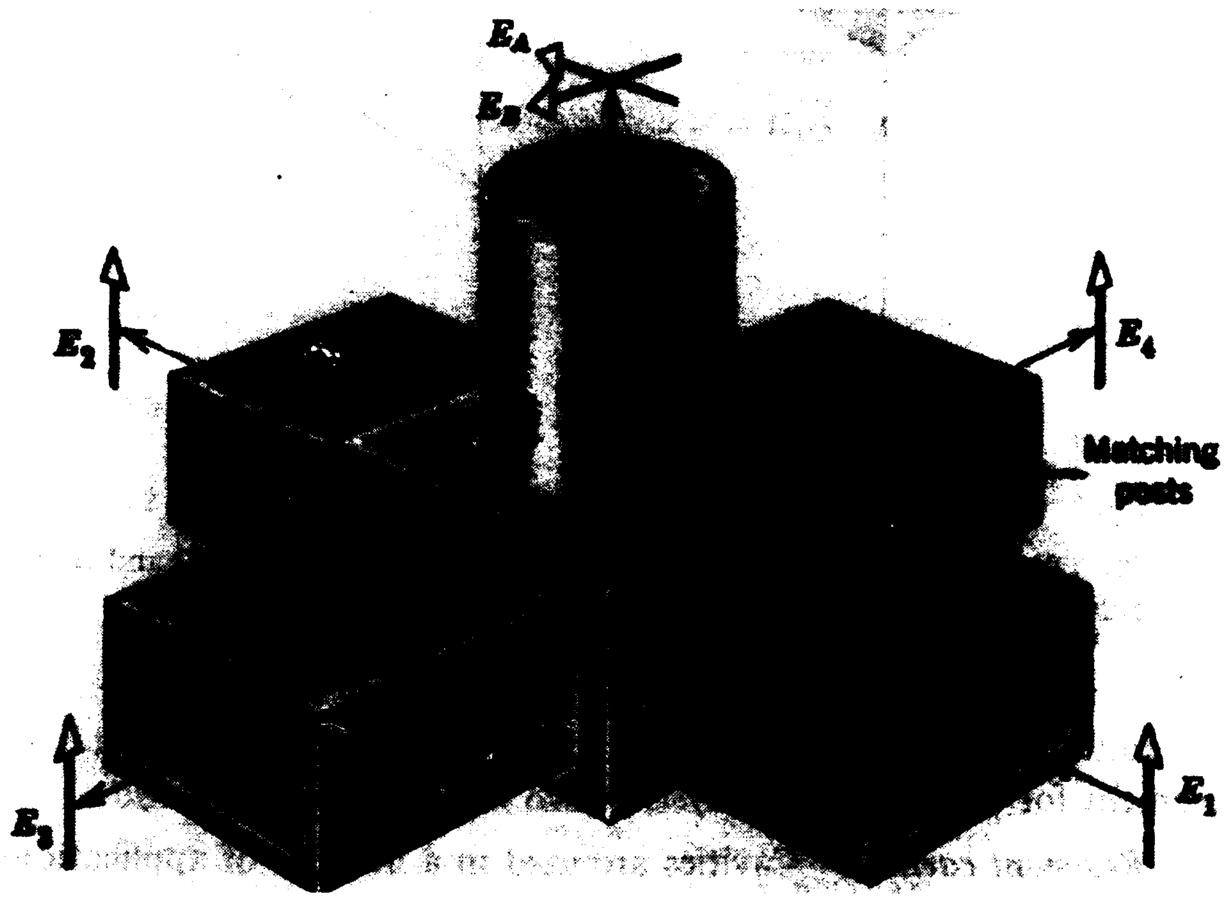


Figure 2.24. Schematic of "turnstile" junction.

This design can be quite compact axially, generates a unidirectional wave, and is capable of high power operation with low loss, but without modifications, it is not suitable for use as the input coupler. Appropriate modifications are suggested by the configuration and performance of an I-J band double-ridged waveguide turnstile coupler built by Dr. Richard True of Litton Electron Devices [14]. The distinguishing aspect of this design, in comparison with a conventional turnstile coupler, is the use of phase-shifted active drive at all four arms.

A major concern is the degree of performance degradation resulting from the replacement of the central matching posts with the beam tunnel. It appears that an acceptable match can be achieved over a reduced frequency range without matching posts, as demonstrated by the measured return loss of the device described above, with and without matching posts (Figs. 2.25-26). Note that the bandwidth of ridged waveguide is inherently greater than that for equivalent smooth rectangular waveguide. Note also that the above device does not include an aperture for electron beam propagation. In the modified design (Fig. 2.27), the lack of the matching posts is at least partially compensated for by bending the rectangular waveguide arms to point towards the calorimeter. Some degree of additional matching is possible with adjustment of the tuning cylinder depth of penetration.

In order to generate a circularly polarized wave with broad instantaneous bandwidth, the excitation scheme is modified. Instead of one active and three passive arms, all arms are actively driven, as in True's device. This requires external microwave circuitry, capable of high-power operation, to generate 0, 90, 180, and 270° phase shifts at the appropriate arms. The degree of circularity across the frequency band is now dependent on a constant phase shift with frequency of the individual external components. An additional benefit of four-port excitation is the distribution of power over four windows instead of one.

A comparison of standard and modified turnstile junction coupler operation is shown in Fig. 2.28. An advantage of the all-active port excitation scheme is the possibility of changing the polarization from LHCP to RHCP to linear or elliptical simply by adjusting the phase and/or amplitude of the driven ports. Launching a TM_{01} mode is also possible with appropriate phasing. Since

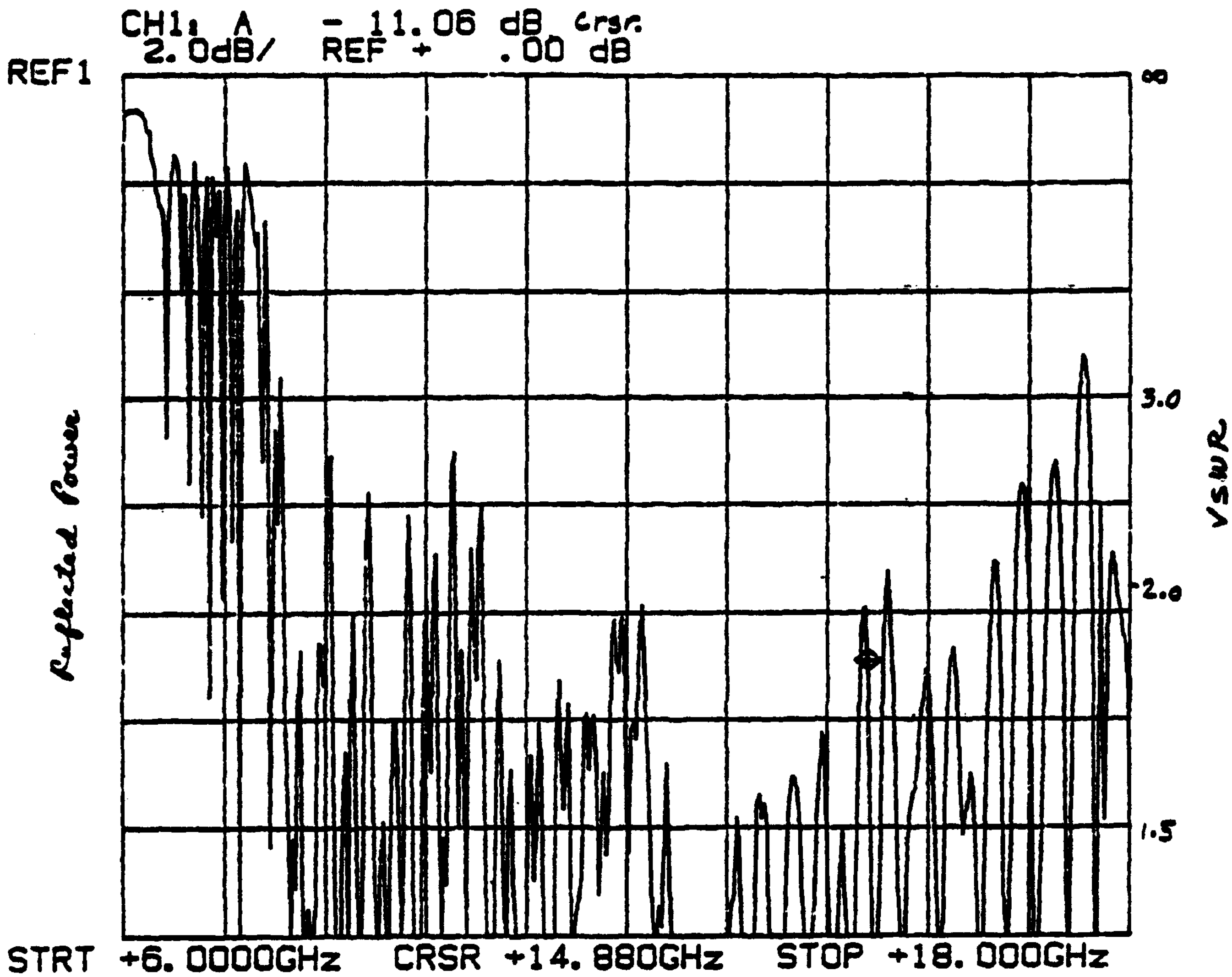


Figure 2.25. Match of magic tee splitter/ridged turnstile combiner combination with original turnstile tri-post tuner setting.

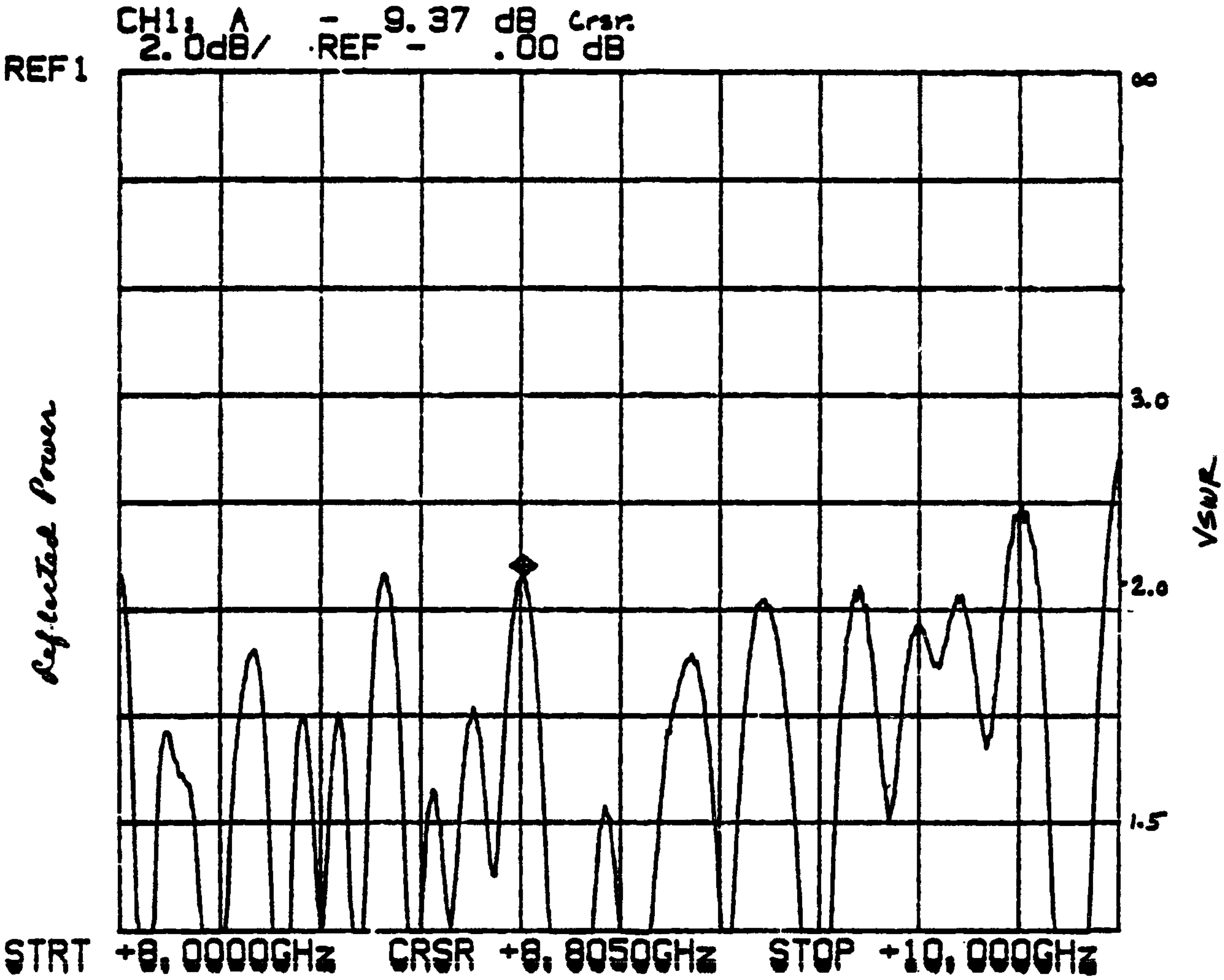


Figure 2.26. Match of magic tee splitter/ridged turnstile combiner combination with turnstile tuning structure removed over reduced frequency range.

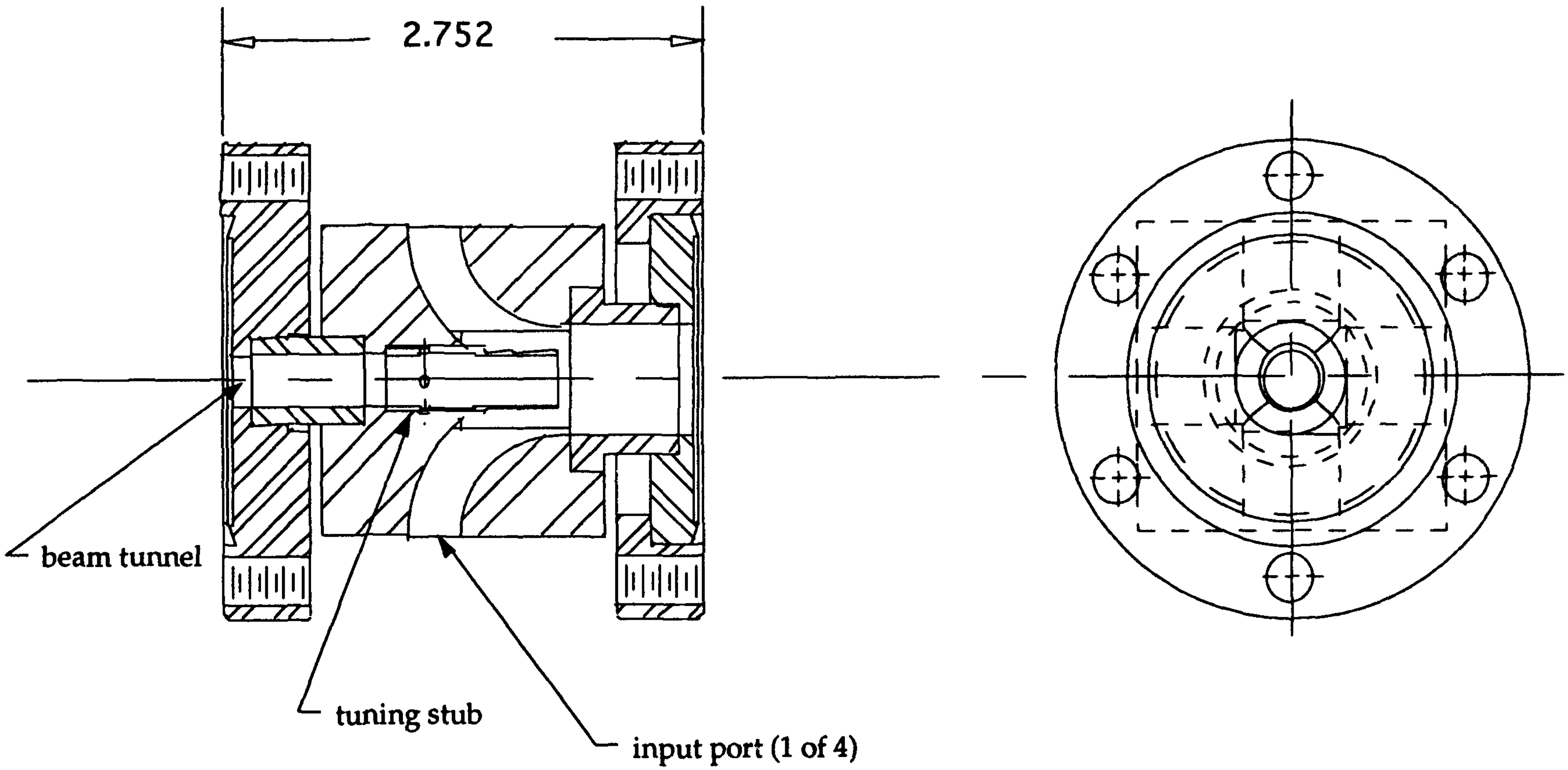


Figure 2.27. Modified turnstile junction input coupler.

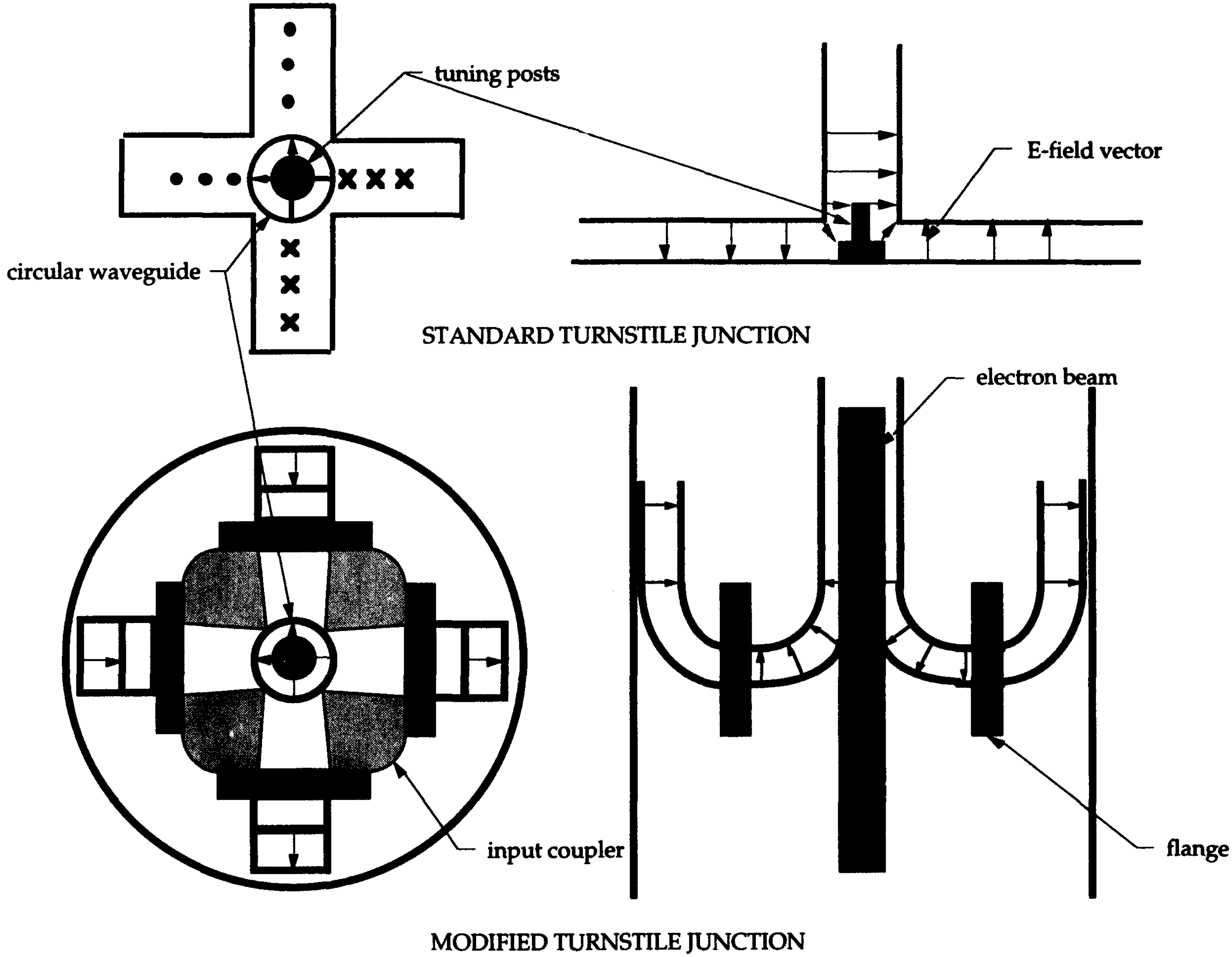


Figure 2.28. Comparison of standard and modified turnstile junction operation.

the actual performance of the modified turnstile coupler is not amenable to analytic computation, the final configuration is determined from cold test measurements.

2.5.3 Mechanical Design.

The general design and fabrication techniques for the input coupler are similar to those for the output monitor (Section 2.6), also a four-port device. Coupler body material is OFHC copper, chosen for easy step-brazing and vacuum compatibility. The modified turnstile junction is fabricated from four sections to be brazed together later, each an identical quadrant containing halves of two curved rectangular waveguide arms. The beam tunnel is drilled through the assembly after the brazing operation. Rectangular guide width in the turnstile section is reduced for cut-off frequency match with the circular guide. Details of each quadrant, without the beam tunnel and tuning cylinder, and the assembled coupler, without windows or flanges, are shown in Fig. 2.29. Vacuum windows for each input port are cut from a sheet of natural mica, 0.005 in. thick, and sealed to the coupler with EPOTEK H-77 hermetic epoxy (service temperature 160°C). In keeping with the modular approach taken for the ubitron, the coupler is a separate section with stainless steel Conflat flanges brazed on each end.

2.5.4 Complete Input Circuit.

As discussed above, the modified turnstile junction input coupler is not functionally complete without external microwave circuitry, specifically components to split the drive signal into the appropriate phases at each input port. For this implementation of the modified turnstile junction coupler, a waveguide 'magic-T' is first used to split the input signal into 0° and 180° components (Fig. 2.30). Each of these components is further split into 0° and 90° components using stripline quadrature hybrid couplers. Thus, the original input signal is split into 0, 90, 180, and 270° components. Depending on the connection sequence at the input coupler ports, either right- or left-circularly polarized waves may be launched. Orthogonal linearly polarized waves may also be launched with 0 and 180° inputs at opposite pairs of ports. Other modes and polarizations may be launched with the same coupler using different amplitudes and phases at the input ports.

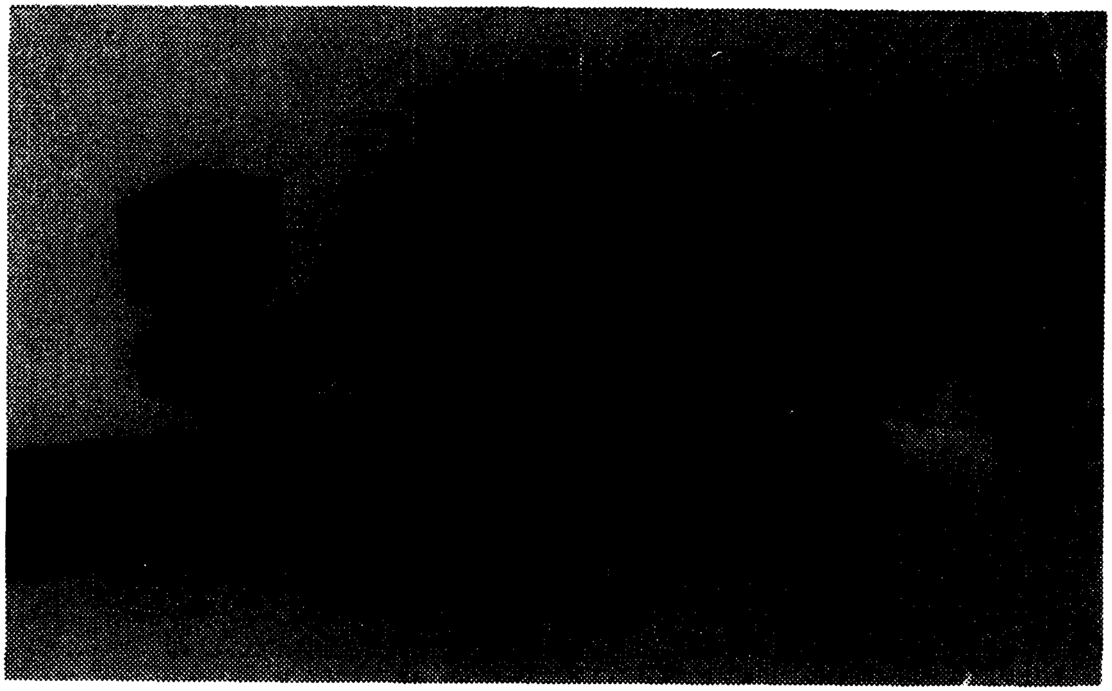


Figure 2.29. Input coupler details.

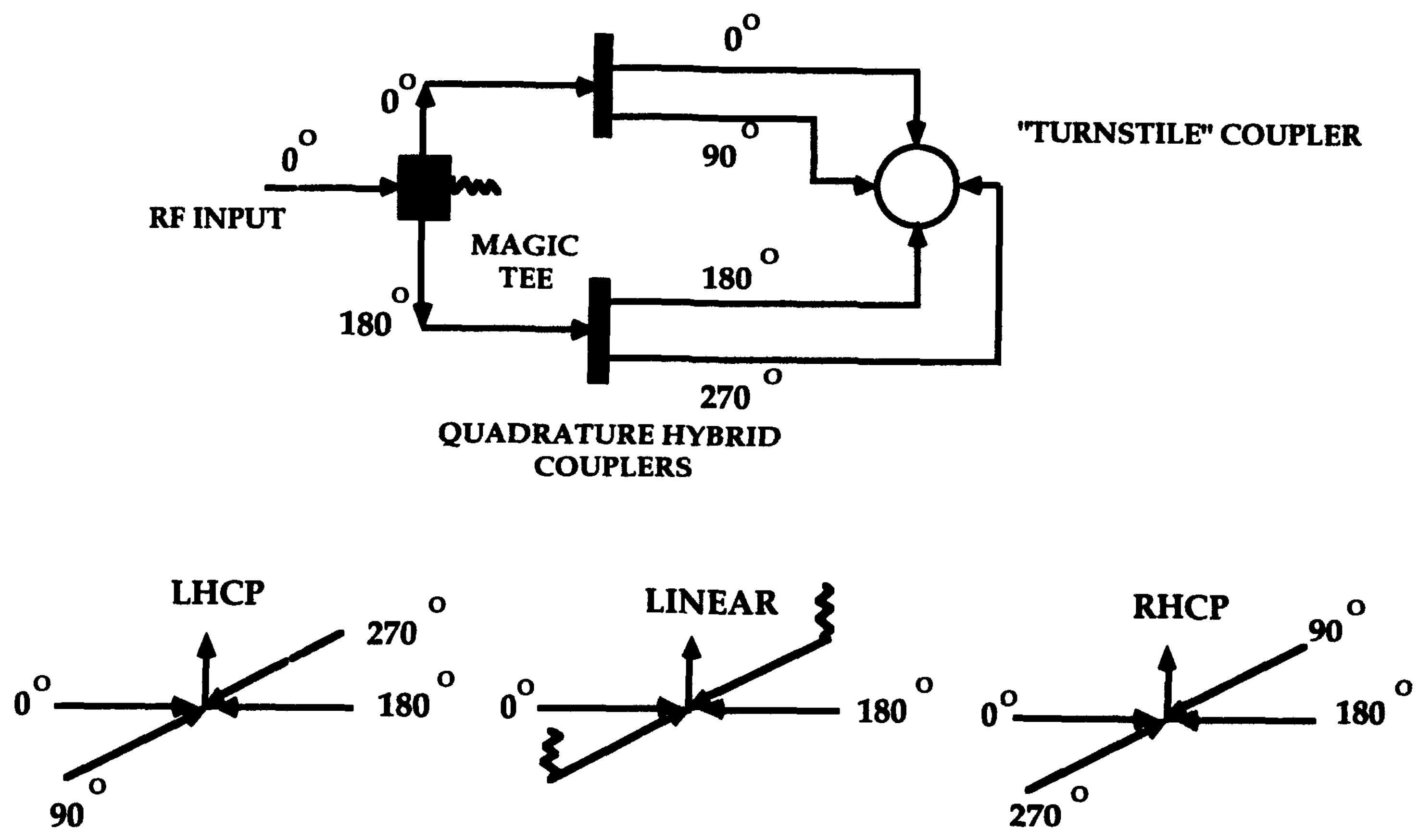
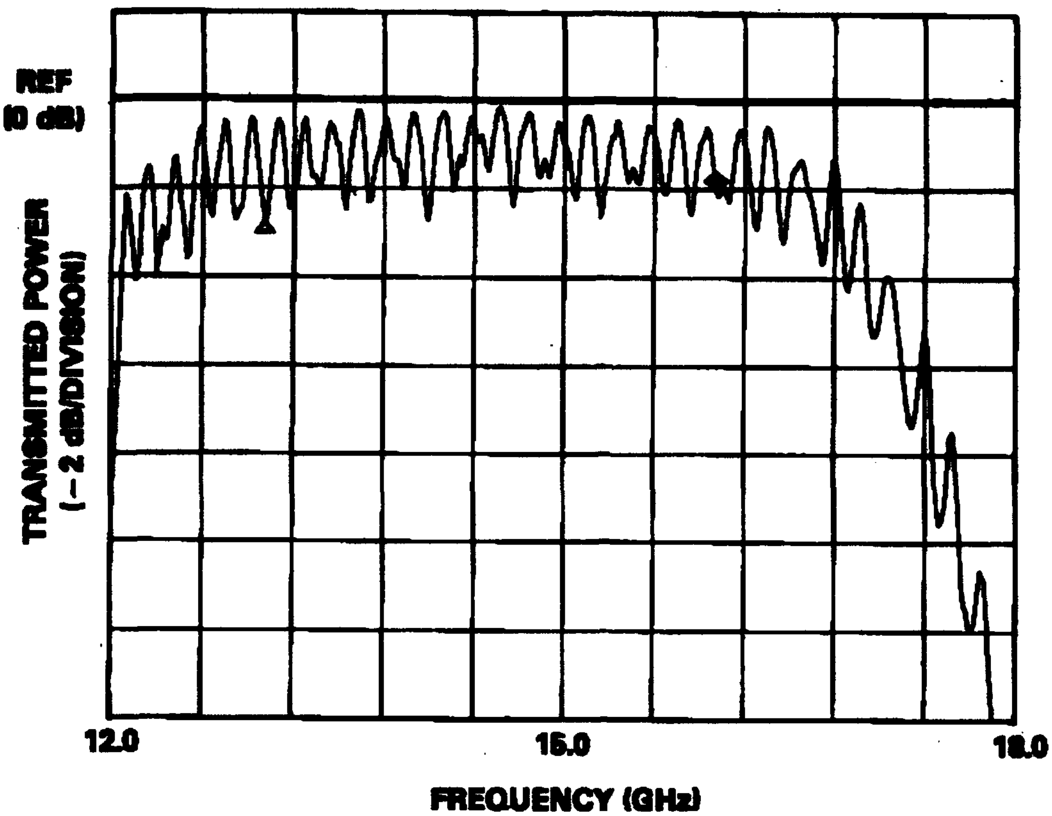


Figure 2.30. Microwave circuit for input coupler.

The final elements of the complete input circuit are the waveguide sections that connect the remotely located phase splitting circuit to the input coupler. The width of each waveguide arm is linearly reduced over the last 6 inches of length from the standard (WR-62) K_u band rectangular guide width of 0.622 in. to 0.548 in. to match impedances between the rectangular and circular waveguides. Each waveguide arm is also terminated at the input end with a custom E-plane miter bend, and at the coupler end with smooth 90° radial bend. Miter bends are required to fit within the solenoid bore, although the VSWR is inferior to that of a smooth bend.

2.5.5 Performance.

As mentioned above, the performance of this coupler design is not amenable to easy computation. In fact, measurements of the circular polarization performance are also difficult. As a consequence, most performance tests have concentrated on linear polarization excitation. Shown in Fig. 2.31 is the measured frequency response of the coupler when excited by 0 and 180° input at opposite ports. Tuning cylinder depth of penetration has been adjusted for best performance for these measurements. The average transmission loss over the expected ubitron operating range (13.5 - 16 GHz) is acceptable, less than 2 dB, but it increases quite rapidly above 16.5 GHz. Isolation between orthogonal ports is approximately 20 dB over most of the normal K_u band frequency range, also shown in Fig. 2.31. This performance is considered acceptable for initial ubitron operation, although the high transmission loss above 16.5 GHz, and, hence, low return loss, could lead to oscillation problems.



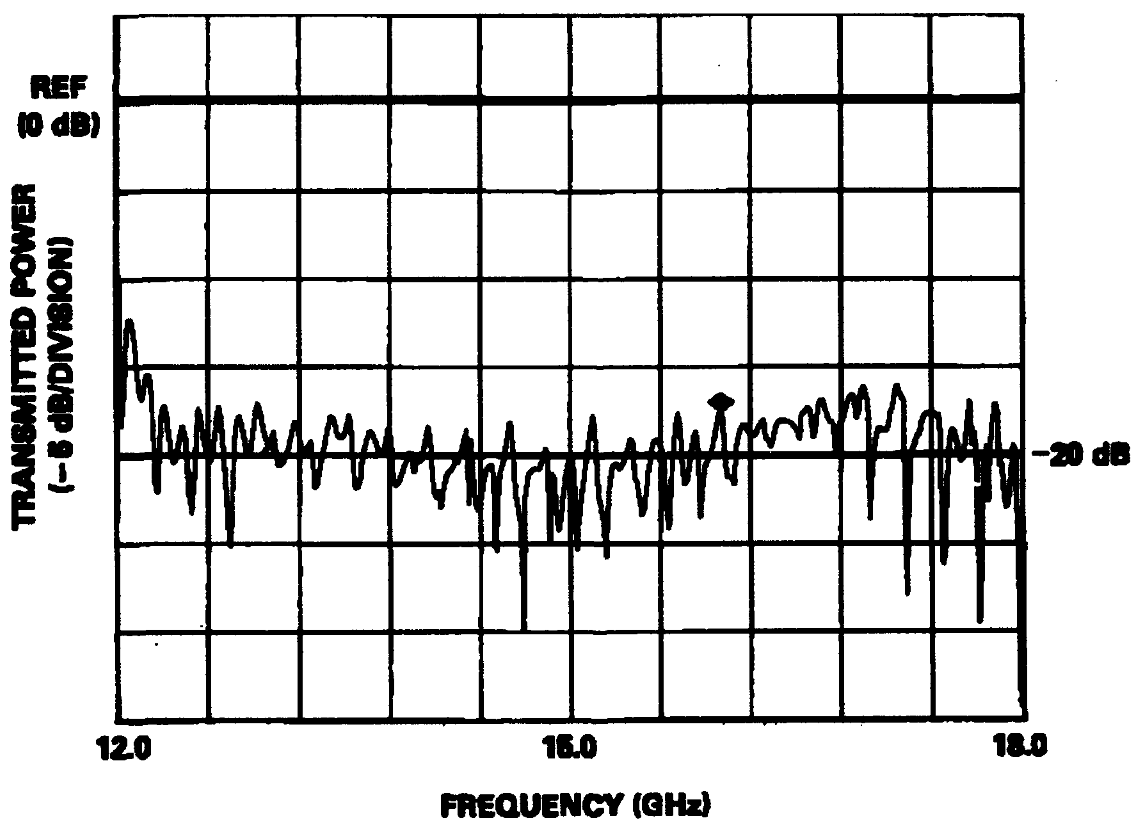


Figure 2.31. Measured transmission loss and isolation of input coupler; linear polarization.

2.6 OUTPUT COUPLER.

2.6.1 Introduction.

The microwave output coupler is one of the primary tools for determining the performance of the ubitron. It is a four-port device that is designed to couple only a small fraction of the circularly polarized TE_{11} radiation into the TE_{10} mode of the coupling rectangular waveguide, using coupling apertures in either the broad or the narrow wall. Wave-shape information obtained from the coupler is used in conjunction with the calorimetric total energy measurement to determine the amplified power output of the ubitron. With a well calibrated attenuation chain, the coupler is also used as a direct power diagnostic. Since the coupler can also respond to other than the desired mode or polarization, significant differences between power levels determined by the two methods can help to identify such modes or polarizations. High directivity is not a requirement for this monitor, since the calorimeter/water load is a well-matched termination.

A general discussion of the significant aspects of output coupler design, construction, and performance follows; details can be found in Ref. 15. Briefly, the design is based on calculations of the electric and magnetic polarizabilities of a small aperture due to a circularly polarized TE_{11} mode in a circular waveguide. The excited field amplitudes of the TE_{10} mode in the coupled rectangular waveguide and, therefore, the coupled power, are calculated from these polarizabilities. Corrections are included for the wall thickness and aperture resonant frequency effects, but not for field variations across the aperture. Aperture locations and spacings are then chosen for optimum performance.

2.6.2 Methodology.

The multi-step design procedure focuses on the selection of the set of parameters that results in the most uniform transfer of power, with frequency, from the fundamental mode in the circular guide to the fundamental mode in the rectangular guide. The following considerations led to the choice of aperture location in the broad-wall of a collinear rectangular waveguide: 1) Rectangular waveguide is chosen to facilitate interfacing with standard waveguide

components, 2) Broad-wall coupling usually results in broader bandwidth performance than narrow wall, 3) The overall radial dimension of the monitor, including any waveguide bends, must be minimized in order to fit within the solenoid bore, and 4) Vacuum windows are somewhat easier to install. Modifications to the basic broad-wall geometry are discussed later.

Given the configuration shown in Fig. 2.32, the following parameters need to be determined: 1) rectangular waveguide width, a', 2) aperture location in broad-wall of rectangular guide, d, and 3) the aperture radius, r_0. The circular waveguide radius, a, has been previously determined by ubitron performance considerations, and the height of the rectangular waveguide is chosen to match that of standard K_u band waveguide. The quantities a', d, and r_0 are first determined using linear polarization input and a Simplex optimization procedure. Since higher order modes in the circular guide may be present, it is desirable that the coupler design preferentially select the fundamental mode. This is accomplished with a series of coupling apertures spaced axially to enhance coupling from the circularly polarized TE_{11} mode and inhibit coupling from the TM_{01} mode. A Simplex procedure is also used for this calculation.

The magnitude of the coupling coefficient is also dependent on the polarization of the incident wave. To assist in mode/polarization discrimination, a narrow-wall adapter was also designed and constructed. The narrow-wall configuration is insensitive to TM modes.

The approach taken here to calculate the coupling coefficient generally follows the procedure given in Collin [16] for calculating the coupling between two rectangular guides. The basic procedure is:

1) Determine the field patterns for the particular modes to be coupled.
2) Calculate the electric and magnetic dipole moments at the aperture location.
3) From the dipole moments calculated above, compute the radiated field amplitudes in the coupled guide.
4) Calculate the coupling coefficient, including any correction factors.

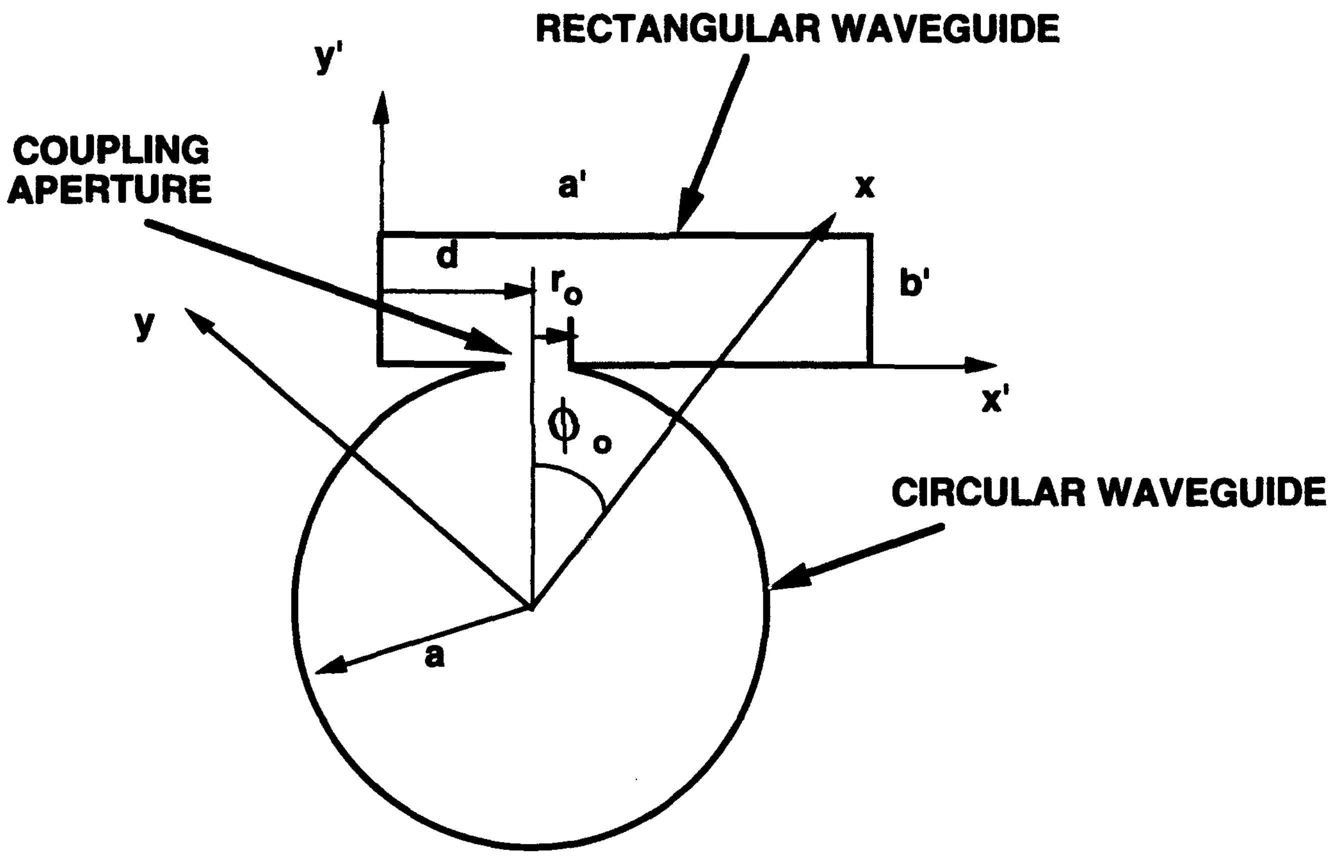


Figure 2.32. Output coupler coupling geometry.

The calculations are based on the Bethe small aperture coupling theory with the following assumptions: 1) the aperture is small in comparison to operating wavelength so that the field is considered constant over the aperture, 2) the aperture is in an infinitesimally thin wall, and 3) the aperture is well removed from any waveguide discontinuities. It is also assumed that the coupling is relatively weak. Detailed knowledge of the exact field structure in the vicinity of the aperture is not required, i.e., the incident wave is not seriously distorted by the aperture. Only the field patterns of a particular mode for each waveguide are required, assuming that the waveguides are unbroken.

The effect of the aperture is represented by electric and magnetic dipole moments in the unbroken infinitesimally thin wall at the location of the aperture. The strength of these moments is proportional to the polarizability (dependent on aperture size and shape) and dependent on the normal electric field and transverse magnetic field at the aperture location. Polarizabilities have been determined for a variety of aperture shapes. The coupling coefficients are calculated from the fields radiated by the electric and magnetic dipoles. A particular mode in the coupled guide will be excited only when either the electric or magnetic field of the mode has a component, at the aperture location, parallel to the electric moment or magnetic moment, respectively.

Correction factors must be applied to account for aperture dimensions not negligibly small in comparison to the operating wavelength, and a finite wall thickness. The resonant frequency effect of large apertures can be approximated by calculating the free-space wavelength at the cutoff frequency for the lowest order mode appropriate to the particular type of excitation in a waveguide having the same cross section as the aperture. The effect of finite wall thickness can be approximated by calculating the attenuation due to a length of transmission line equal to the wall thickness and diameter equal to the aperture diameter. The calculations also account for differing waveguide impedances for circular and rectangular waveguides. An additional correction factor could also be included to account for the field variation across the aperture, but this is not considered significant in this case.

2.6.2.1 Basic Coupling Equations.

2.6.2.1.1 Broad Wall Coupling.

2.6.2.1.1.1 Linear Polarization.

For the ubitron output coupler design, it is assumed that a unit amplitude, unidirectional TE_{11} wave propagates down the circular guide, which is coupled to a bidirectional TE_{10} wave in the rectangular guide. The power in each mode is:

$$P_{11} = \frac{1}{2} U_{11}^2 \frac{\beta}{\omega\mu}$$

$$P_{10} = \frac{1}{2} U_{10}^2 \frac{\beta_{10}}{\omega\mu} ,$$

where U_{11} and U_{10} are the amplitudes of the TE_{11} and TE_{10} modes and β and β_{10} are the propagation constants in the circular and rectangular guides, respectively. The coupling coefficient is

$$C = 10 \cdot \log\left|\frac{P_{10}}{P_{11}}\right| = 10 \cdot \log\left|\frac{U_{10}^2 \beta_{10}}{U_{11}^2 \beta}\right| .$$

When corrected for finite wall thickness and large aperture effects, the final expression for coupling between a linearly polarized TE_{11} circular waveguide mode and the TE_{10} rectangular waveguide mode is

$$C = 20\log\left[\frac{0.2433 r_0^3}{a\beta_{10}\sqrt{a'b'}}\left|\left(\frac{\omega^2}{c^2} CF_e - 2.0025\beta\beta_{10} CF_m\right)\sin\frac{\pi d}{a'}\sin\phi_0 + \left(\frac{21.2519}{aa'} CF_m\right)\cos\frac{\pi d}{a'}\cos\phi_0\right|\right] + 10\log\left|\frac{\beta_{10}}{\beta}\right| ,$$

where:

$$CF_{e(m)} = \exp\left\{\frac{\dfrac{-2\pi A_{e(m)}t}{\lambda_{ce(m)}}\sqrt{1-\left(\dfrac{\lambda_{ce(m)}}{\lambda}\right)^2}}{1-\left(\dfrac{\lambda_{ce(m)}}{\lambda}\right)^2}\right\},$$

$\lambda_{ce} = 2.613\sqrt{\varepsilon_r}r_0$
$\lambda_{cm} = 3.412\sqrt{\varepsilon_r}r_0$
$A_e t = 1.0103t + 0.0579r_0$
$A_m t = 1.0064t + 0.0819r_0$
$\varepsilon_r =$ relative permittivity
$t =$ wall thickness
$r_0 =$ aperture radius

2.6.2.1.1.2 Circular Polarization.

To calculate the coupling between a unit amplitude circularly polarized TE_{11} mode and the TE_{10} mode, the field expressions must be modified to describe a circularly polarized wave. The condition for circular polarization is that the x and y components are of equal magnitude, but 90° apart in time phase. The relevant field components of a TE_{11} circularly polarized wave are

$$E_{r_c} = \frac{\sqrt{2}}{2}\left[E_r(\phi_0) + jE_r\left(\phi_0 + \frac{\pi}{2}\right)\right]$$
$$\vec{H}_{tw_c} = \frac{\sqrt{2}}{2}\left[\vec{H}_{tw}(\phi_0) + j\vec{H}_{tw}\left(\phi_0 + \frac{\pi}{2}\right)\right].$$

Including the appropriate correction factors for wall thickness and aperture size, the coupling factor between a circularly polarized TE_{11} wave and a TE_{10} wave is

$$C = 20\log\left[\frac{0.2433r_0^3}{a\beta_{10}\sqrt{2a'b'}}\left\{\left(\frac{\omega^2}{c^2}CF_e - 2.0025\beta\beta_{10}CF_m\right)^2\sin^2\frac{\pi d}{a'} + \left(\frac{21.2519}{aa'}\right)^2\cos^2\frac{\pi d}{a'}\right\}^{1/2}\right] + 10\log\left|\frac{\beta_{10}}{\beta}\right|,$$

where CF_e and CF_m are defined above.

2.6.2.1.2 Narrow Wall Coupling.

To assist in TE/TM mode determination, a narrow wall adapter was constructed that does not couple to TM modes. Using the above methodology, the following relevant coupling factors have been computed:
linear TE_{11} to TE_{10} (narrow wall):

$$C = 20\log\left\{\frac{5.1703 r_0^3 C F_m}{\beta_{10} a^2 a' \sqrt{a'b'}} \cos\phi_d\right\} + 10\log\left|\frac{\beta_{10}}{\beta}\right|$$

circular TE_{11} to TE_{10} (narrow wall):

$$C = 20\log\left\{\frac{3.656 r_0^3 C F_m}{\beta_{10} a^2 a' \sqrt{a'b'}}\right\} + 10\log\left|\frac{\beta_{10}}{\beta}\right|.$$

2.6.2.2 Multi-aperture Coupling.

Although only coupling from the TE_{11} mode is desired, an unfortunate consequence of the choice of the broad-wall coupling geometry is the possibility of significant coupling from the TM_{01} mode in the circular waveguide (f_{co} = 14 GHz). Two mitigating factors are: 1) the TM_{01} mode should not be excited by the input coupler, and 2) the TM mode is not amplified by the ubitron mechanism. The expected gain of the ubitron operating in the TE_{11} mode is 25 - 30 dB. Even assuming equal powers in the TE_{11} and TM_{01} modes before amplification, the input power of the TM mode to the output coupler will be down by 25 - 30 dB in comparison to the TE mode, unless there is another gain mechanism.

However, to insure that only the TE_{11} mode is coupled, the basic design is modified to a mode selective design by the addition of appropriate apertures. That is, the additional apertures will enhance coupling from the TE_{11} mode while suppressing coupling from the TM_{01} mode. Directivity is not a design goal, since the water load/calorimeter is sufficiently well matched to minimize reflections.

The multi-aperture output coupler geometry consists of two pairs of apertures which are located such that the spacing of the first pair results in a directional coupler, and the spacing of the second pair relative to the first results in suppression of the undesired mode. For n apertures, the resulting coupling for the TE_{11} mode is [17]:

$$C_{11} = 20\log\left|\sum_{q=0}^{n-1}\left[-j\left(s_q\beta + (s_{n-1} - s_q)\beta_{10}\right)\right]\right| + 20\log\kappa_{11}\ .$$

Assuming equal amplitudes for both modes, the suppression of the TM_{01} mode is:

$$C_{suppr} = 20\log\left|\frac{\sum_{q=0}^{n-1}\left[-j\left(s_q\beta + (s_{n-1} - s_q)\beta_{10}\right)\right]}{\sum_{q=0}^{n-1}\left[-j\left(s_q\beta_{01} + (s_{n-1} - s_q)\beta_{10}\right)\right]}\right| + 20\log\frac{\kappa_{11}}{\kappa_{01}}$$

Here, κ_{11}, κ_{01} and β, β_{01} refer to the respective TE_{11} and TM_{01} single aperture coupling factors and propagation constants.

2.6.2.2.1 Parameter Optimization.

The parameters of the basic coupling equations are to be chosen for the flattest TE_{11} frequency response and the most TM_{01} mode suppression in the range of 14 to 16 GHz. Following is the list of parameters that determine the output coupler performance:

a	circular waveguide radius
a',b'	rectangular waveguide dimensions
r_0	aperture radius (for all apertures)
d	position of aperture(s) in broad-wall
s_{0-3}	axial positions of apertures

Since the waveguide radius, a, and the rectangular guide height, b', are fixed, the initial parameters to be determined are the rectangular guide width, a', the location of the apertures in the broad-wall, d, and the aperture radius, r_0 (see Fig. 2.32, including narrow wall coupling geometry). These parameters are

chosen for maximum TE_{11} mode coupling flatness, with no consideration to TM_{01} mode suppression. The final parameters to be determined are the aperture axial positions, chosen to maximize TE_{11} coupling while minimizing TM_{01} coupling. To help in the parameter determination, an optimization routine based on the Simplex algorithm was utilized. The particular Pascal implementation of this algorithm is due to Caceci and Caceris [18].

In the first phase, the algorithm searches for the combination of a', d, and r_0 that, within certain tolerance factors, minimizes the difference between the calculated coupling factor and the desired constant coupling factor. No constraints are placed on these parameters for the optimization. For the second phase, the quantity to be optimized is the difference between the TE_{11} and TM_{01} couplings, with the aperture locations, s_0, s_1, s_2, s_3, as the optimization parameters. The only constraint is that the overall axial extent of the apertures must be on the order of 40 mm.

Based on the first phase optimization results, the set of parameters chosen for the initial output coupler design is: a' = 19.29 mm, d = 7.48 mm, and r_0 = 1.44 mm. In the actual output monitor, the broad-wall dimension is linearly reduced to the standard K_u band waveguide width of 15.8 mm over a 25 mm length. A commercial resistive load is placed at the upstream end of the coupled guide to absorb the coupled wave in the unwanted direction. Based on the second phase optimization results, the final aperture locations are: s_0 = 0, s_1 = 15.4, s_2 = 19.7, and s_3 = 34.9 mm. In this case, the coupling from the TE_{11} mode is ~-60 dB. The same aperture locations are used with the narrow wall adapter, which is attached to one of the broad-wall ports in place of its cover plate. Coupling from the TE_{11} mode is increased to ~-50 dB. The performance of the coupler is discussed below.

2.6.3 Mechanical Design.

The general design and fabrication requirements for the output coupler are as follows:

1. Nonmagnetic materials are to be used throughout.
2. The vacuum envelope will incorporate conventional wisdom regarding pumping speed, virtual leaks, bakeout temperatures, and mechanical stresses.
3. The output waveguides will be oriented radially.
4. The entire assembly must fit through the four-inch bore of the solenoid.

The material chosen for the bulk of the coupler is OFHC copper, for easy step-brazing and vacuum compatibility. In keeping with the modular approach taken for the ubitron, the coupler is a separate section with stainless steel Conflat flanges brazed on each end. A particular consequence of Requirement (2) is the choice of vacuum window location. Separate vacuum windows are placed and sealed over each coupling aperture in order to prevent the severe reduction in pumping speed associated with pumping through small orifices. Each window is cut from a sheet of natural mica, 0.005 in. thick, and sealed to the coupling structure with EPOTEK H-77 hermetic epoxy (service temperature 160°C). This arrangement is also mechanically much stronger than the alternative of supporting the same window thickness over the full K_u waveguide area.

The transition of the rectangular coupling waveguide orientation from parallel to the circular waveguide axis to the all-perpendicular "turnstile" orientation of four symmetric radial branches is the most difficult design/assembly problem. The solution is to separate the coupler into two major sections, the coupling section and the turnstile block assembly. Construction details of the coupler and narrow wall adapter are shown in Figs. 2.33-34, respectively. A photograph of the completed output coupler, without windows or adapter, is shown in Fig. 2.35. Also shown are the separate coupling and turnstile block sections.

As the final step in the assembly sequence, separate vacuum windows are fastened and sealed over each coupling aperture. Each window is cut from a sheet of natural mica, 0.005 in. thick, and sealed to the coupling structure with EPOTEK H-77 hermetic epoxy. This epoxy has a rated service temperature of

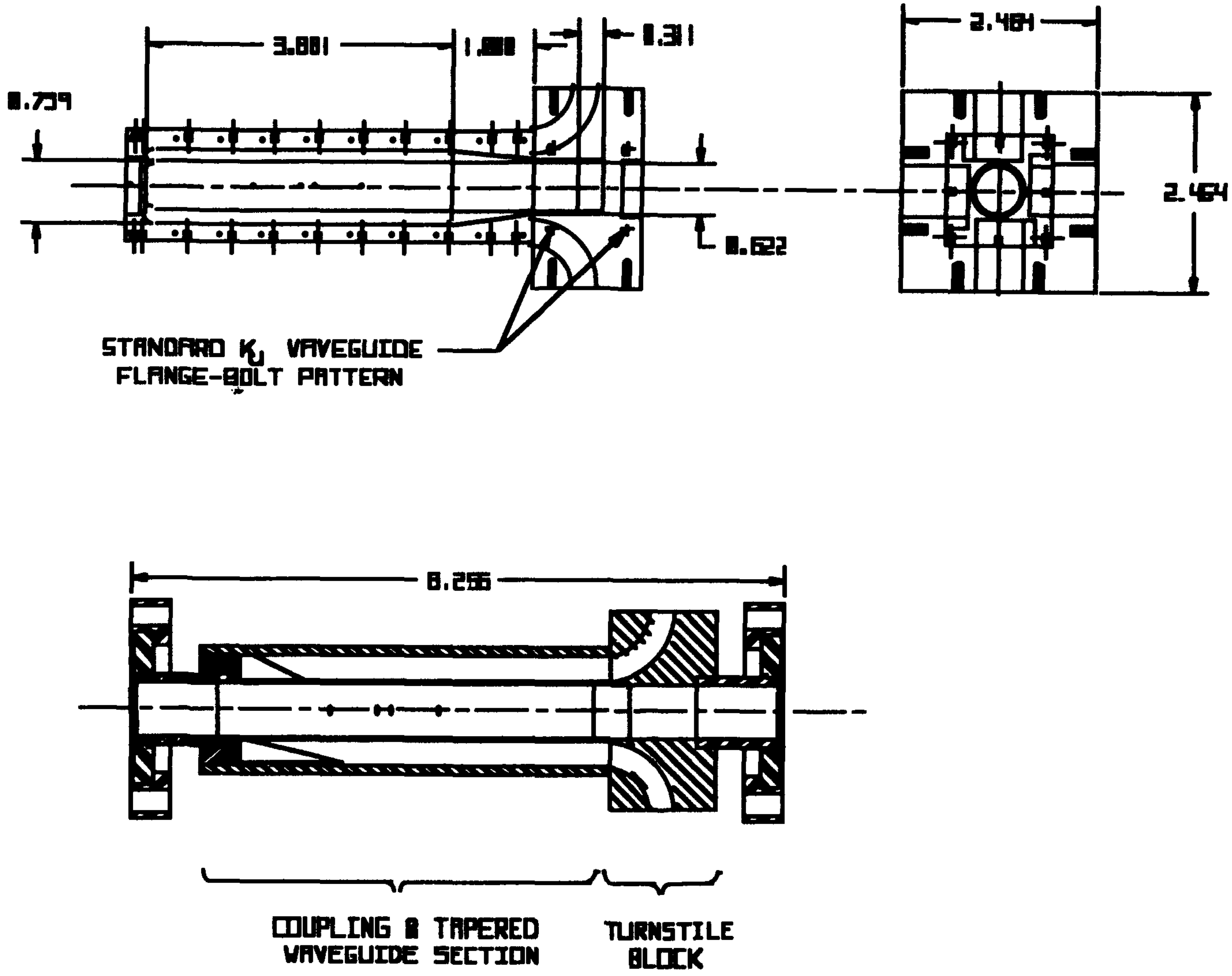


Figure 2.33. Output coupler mechanical design.

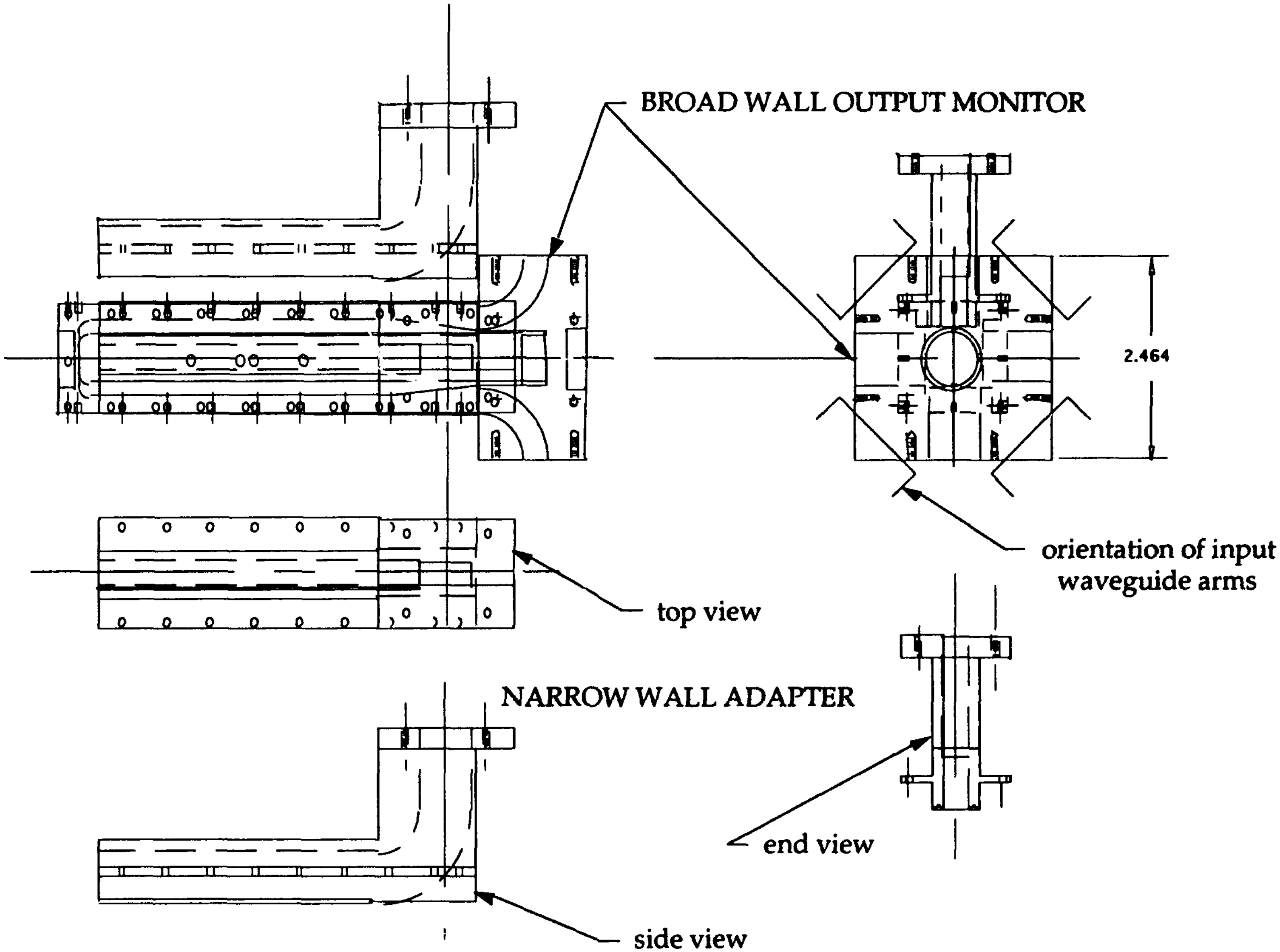


Figure 2.34 Mechanical design of narrow wall adaptor.

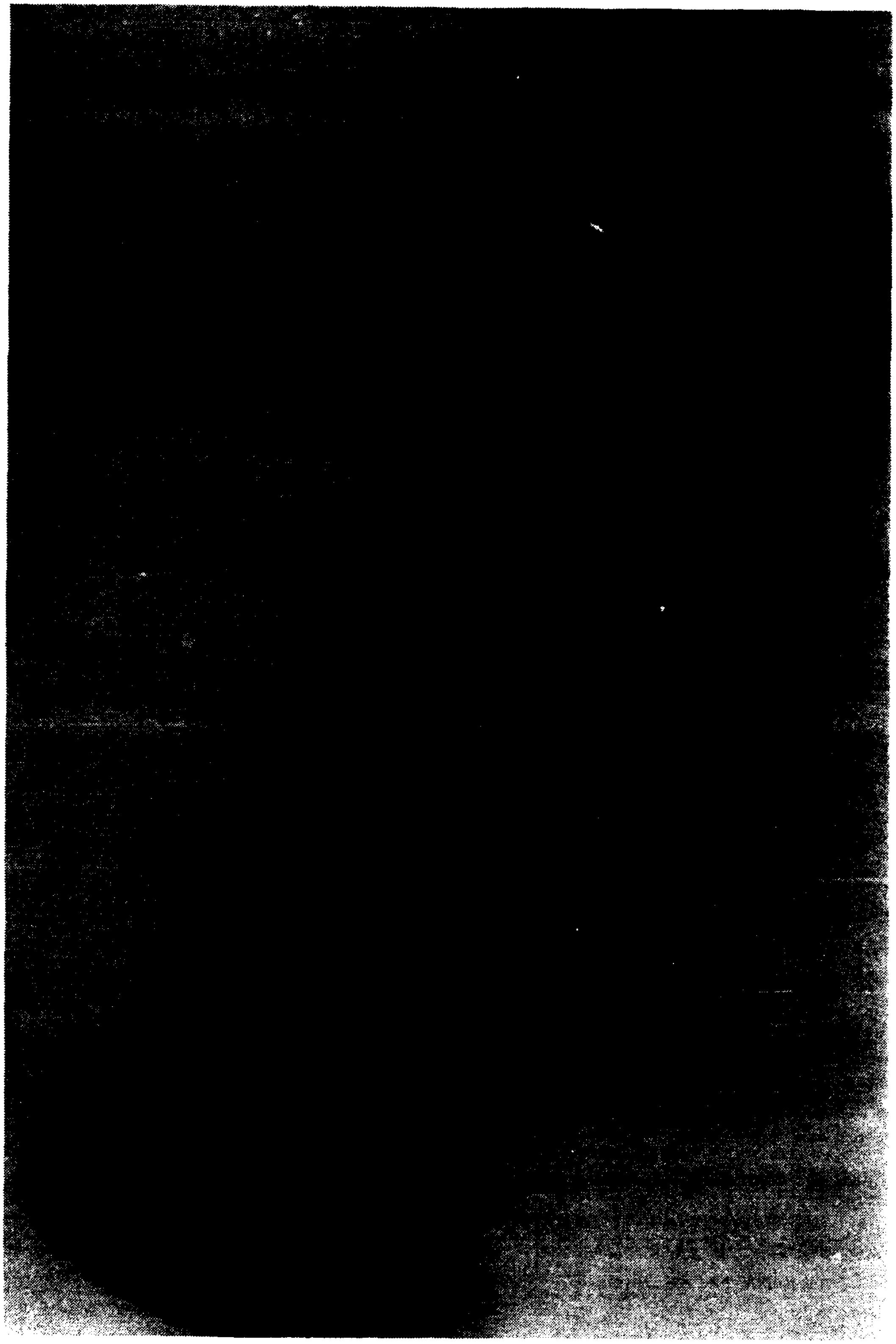


Figure 2.35. Completed output coupler.

160°C. The resistive load is fastened to the coupling structure with conducting epoxy. Performance details of the completed coupler are given below.

2.6.4 Performance.

Due to the small coupling coefficient, the output coupler was calibrated using an amplifier by comparison with a calibrated 50-dB directional coupler. Measurements were made with the coupler in its final configuration, i.e., with 5 mil mica vacuum windows, Conflat flanges, and terminated with the water load/calorimeter.

Although the agreement between measurement and calculation is, in general, good, two points should be made when comparing the measurements with calculations. 1) Due to fabrication difficulties, a port-to-port response variation resulted from the window assembly procedure, whereby some of the coupling apertures were partially filled by the sealing epoxy. Exact computation of the expected coupling coefficient is difficult due to the non-uniform aperture filling. However, the envelope of the port responses is within the bounds of coupling without a dielectric and coupling assuming that all apertures are uniformly filled with a dielectric with relative permittivity on the order of two. Only the two ports with the least epoxy filling are used, one in the broad-wall configuration, and the other with the narrow-wall adapter attached. 2) Depending on the tube assembly procedure, impedance mismatches can occur at the Conflat joints. This is usually manifested by a sharp drop in transmitted power in the vicinity of 17.8 GHz.

Comparisons of the measured and calculated coupling factor between the linearly polarized TE_{11} mode in the circular waveguide and the TE_{10} mode in the rectangular guide for broad-wall coupling are shown in Fig. 2.36. Note that two excitation extremes are shown, E-field transverse and perpendicular to the coupling apertures. The coupling factors for circularly polarized TE_{11} input are shown in Fig. 2.37. Note that the manufacturer's specifications for 3-dB bandwidth of the circular polarizer is approximately 12.6 to 15.5 GHz. The agreement between measurement and calculation is seen to be especially good in this case.

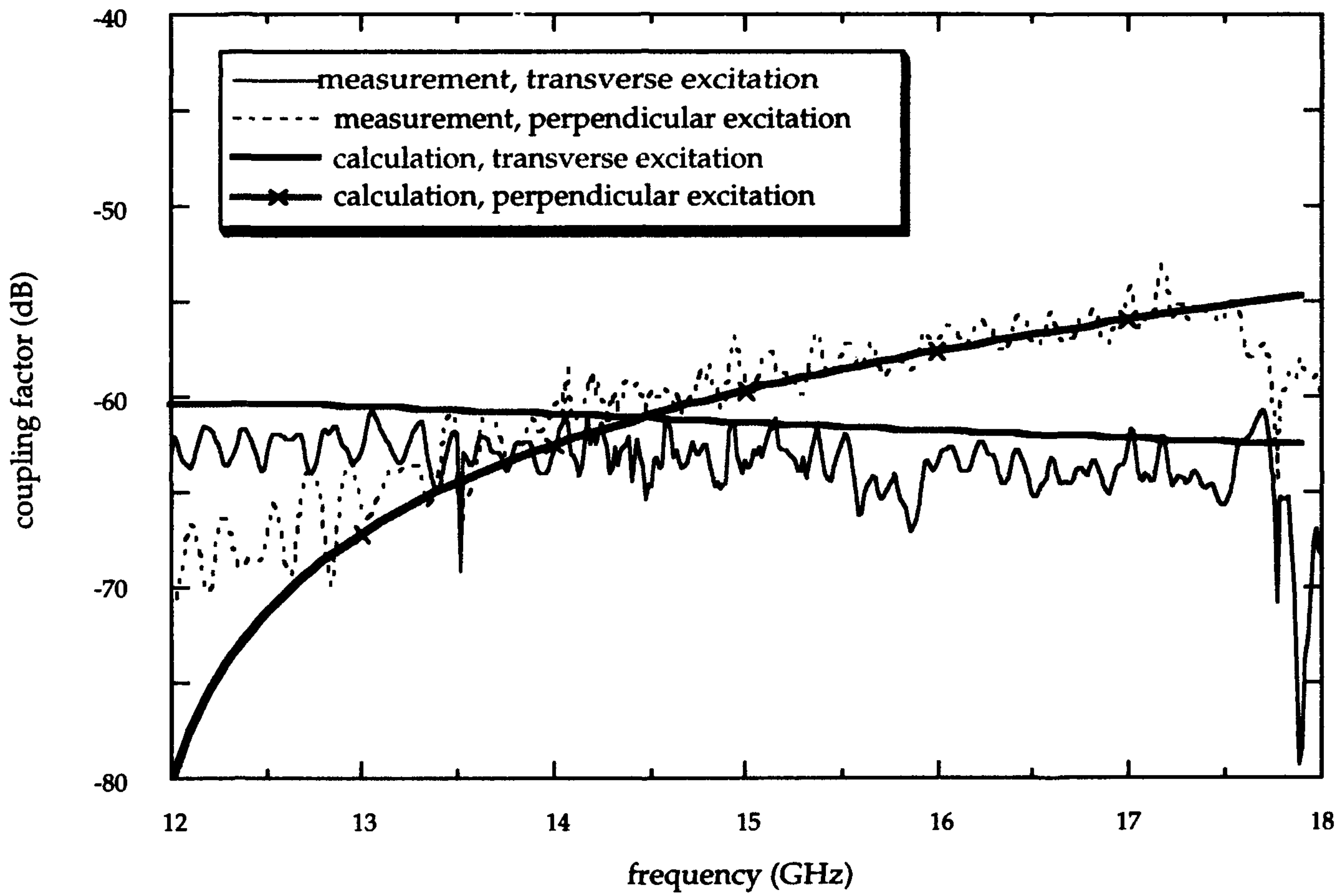


Figure 2.36. Measured and calculated broadwall coupling factors; linear polarization.

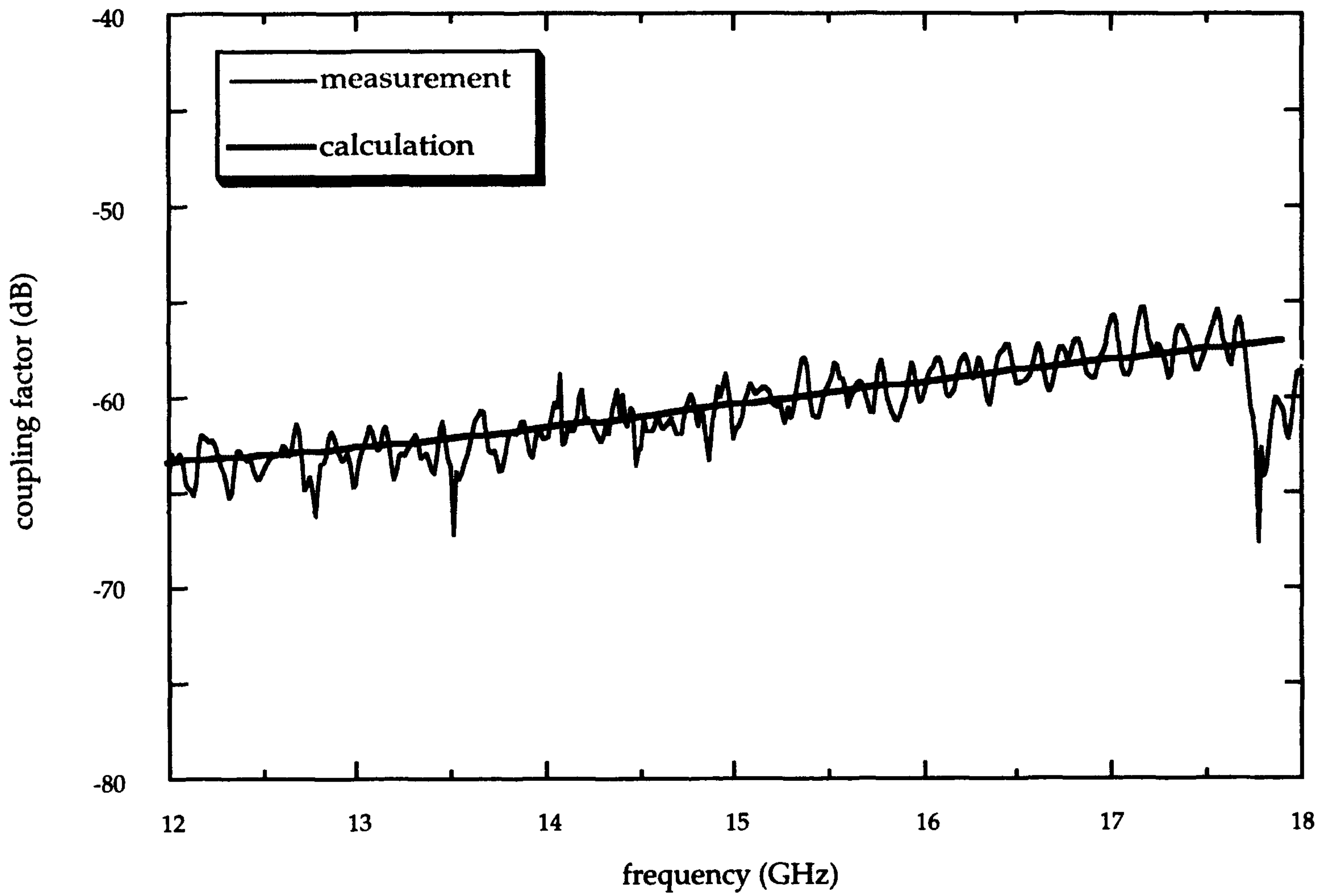


Figure 2.37. Measured and calculated broadwall coupling factors; circular polarization.

Results of the measurements using the narrow-wall adapter for transverse linear and circularly polarized TE_{11} excitation are shown in Figs. 2.38-39, respectively. Note that there should be no narrow-wall coupling for perpendicular excitation. The agreement between measurement and calculation is expected to be slightly worse in this case due to a higher percentage of epoxy in the coupling apertures. The effect of epoxy in the coupling apertures is more explicitly demonstrated in Fig. 2.40, which shows the port coupling factor variation with circular polarization excitation. The narrow-wall adapter is attached to Port 42.

As a cross-check of the output coupler and calorimeter calibrations (Section ?.7), both diagnostics were used with the ubitron fully assembled. The ubitron microwave input circuit, including the input coupler and the HPA, in effect replaced the network analyzer and commercial polarizer. The input power transmitted through the system was then measured using the previously determined calorimeter calibration. The output coupler coupling factor is computed from the power measured at the Wavetek Peak Power Meter detector and the power measured with the calorimeter, $C = 10\log(P_{ppm} / P_{cal})$. Both a 20-W CW amplifier and the repetitively-pulsed HPA were used for these measurements. As shown in Fig. 2.41, the agreement between calculation and the network analyzer and calorimeter techniques is fairly good, resulting in added confidence in both diagnostic calibrations.

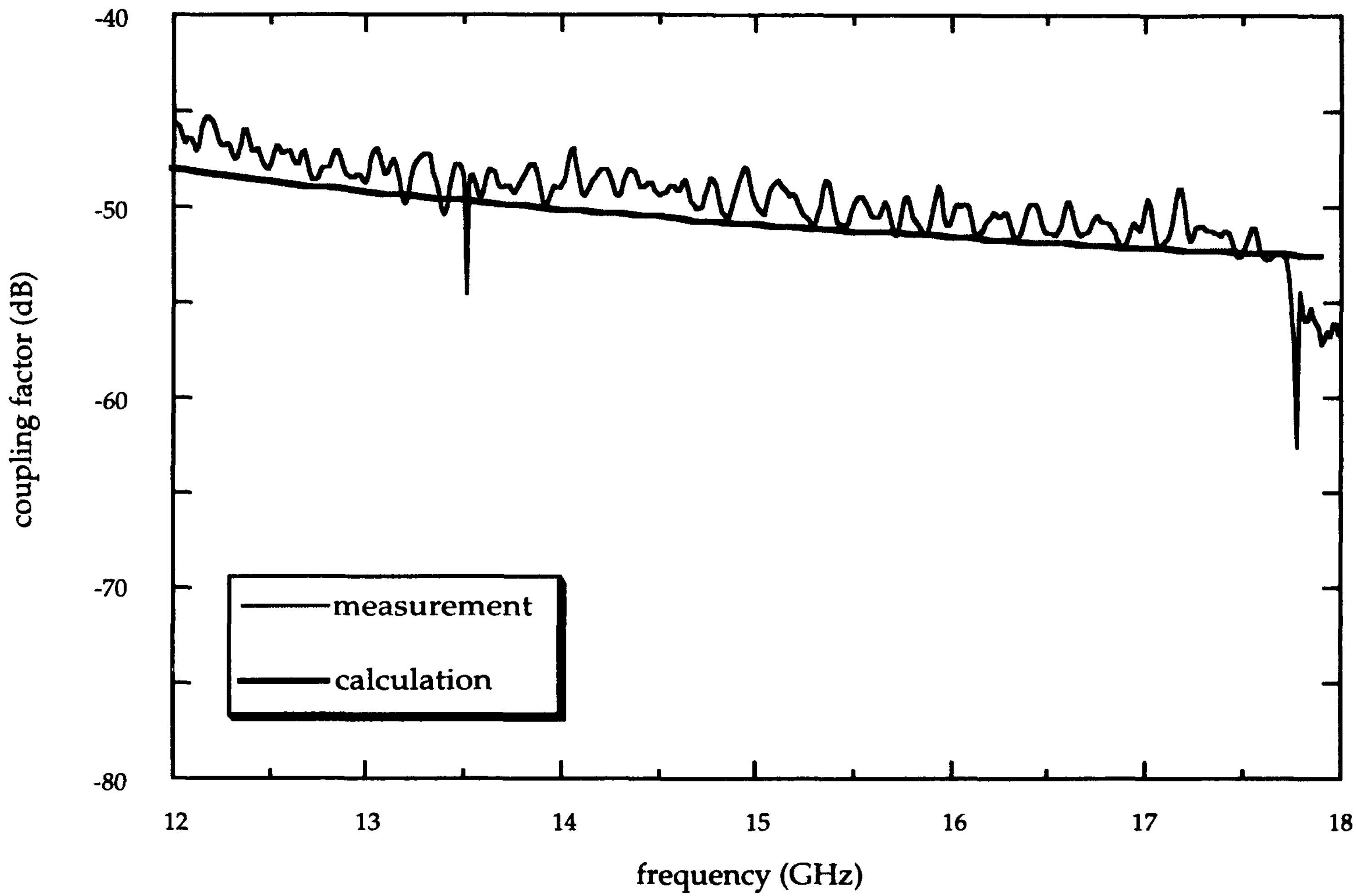


Figure 2.38. Measured and calculated narrow wall coupling factors; linear polarization.

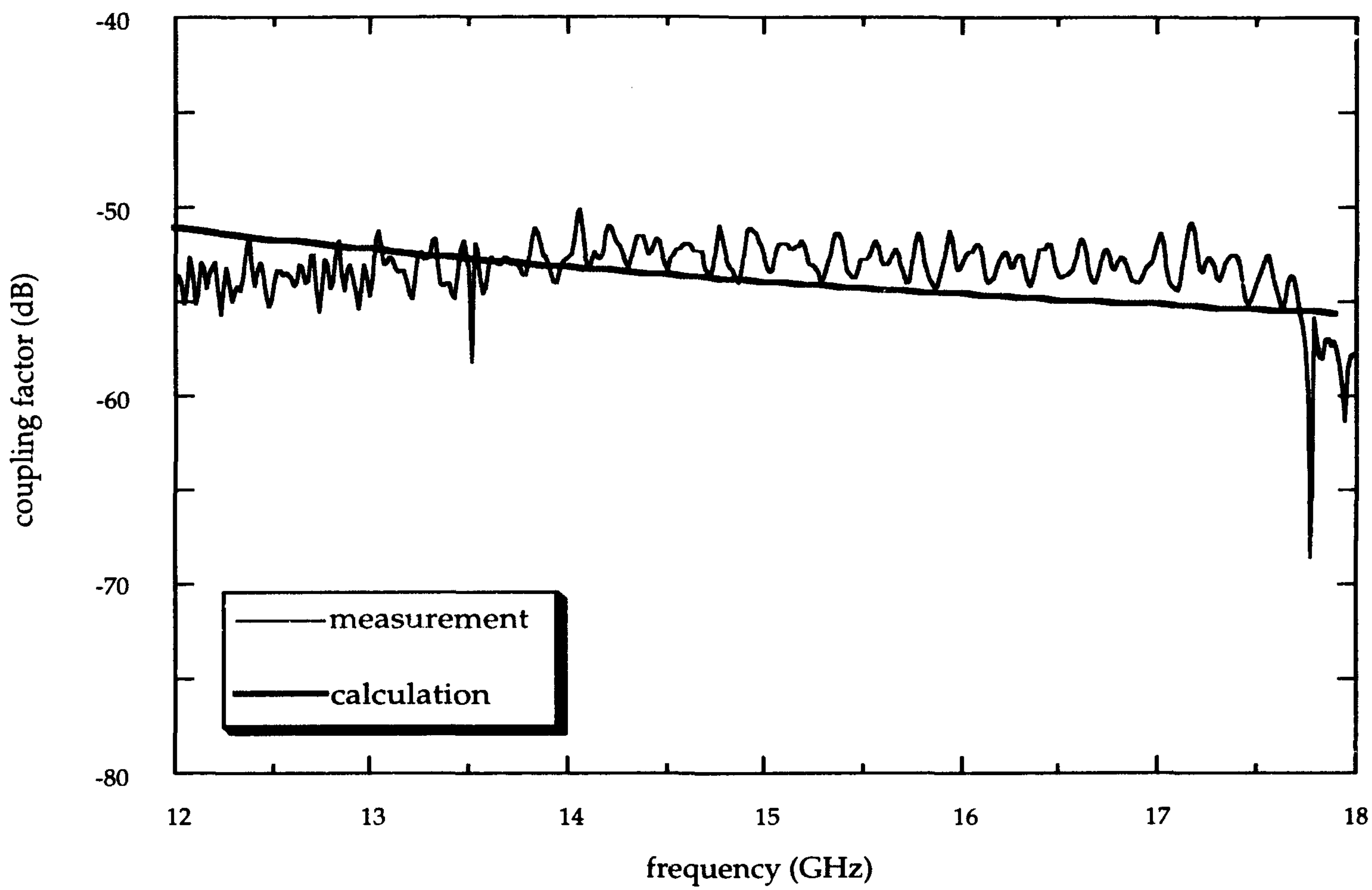


Figure 2.39. Measured and calculated narrow wall coupling factors; circular polarization.

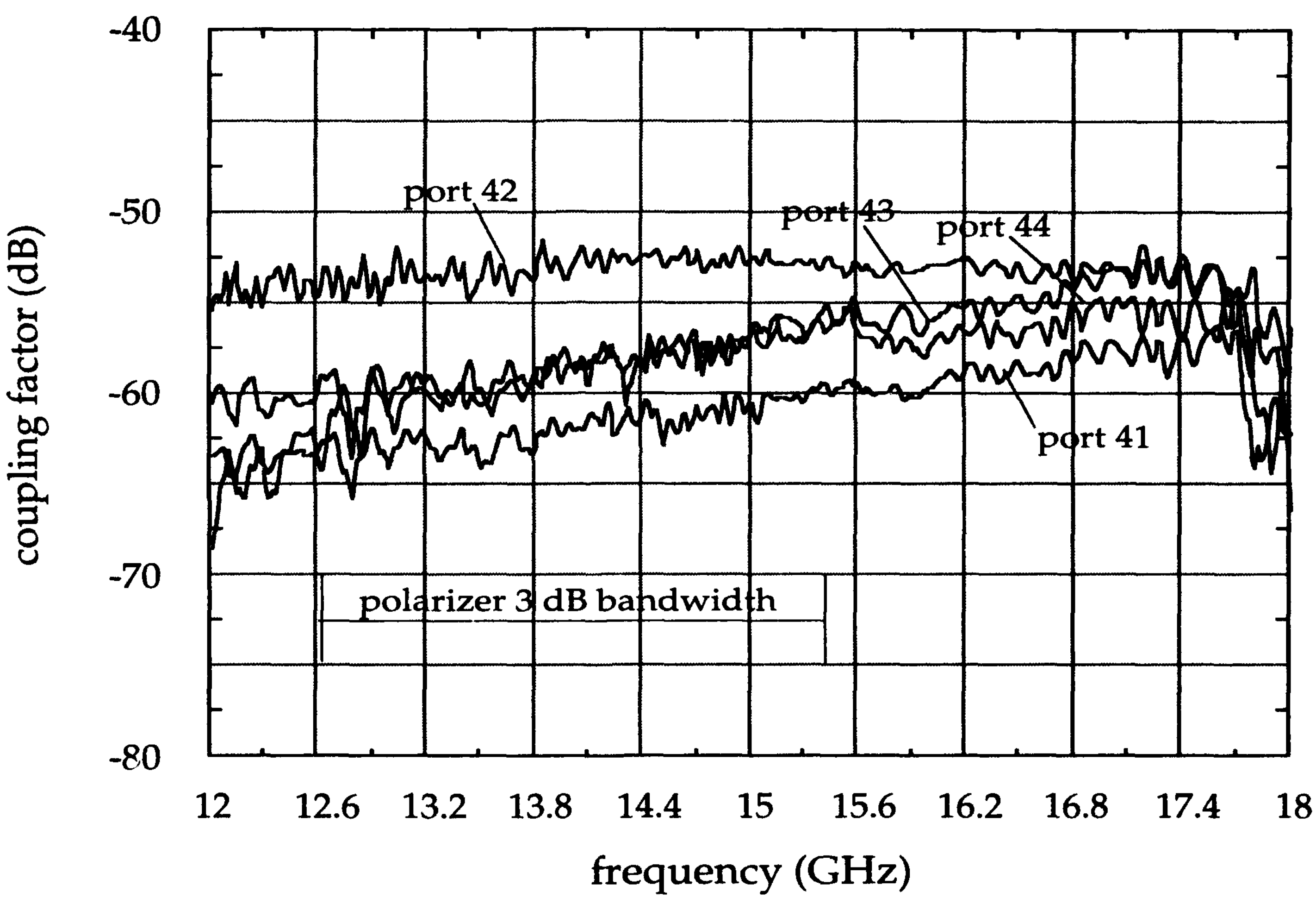


Figure 2.40. Coupling factor variation due to epoxy; circular polarization.

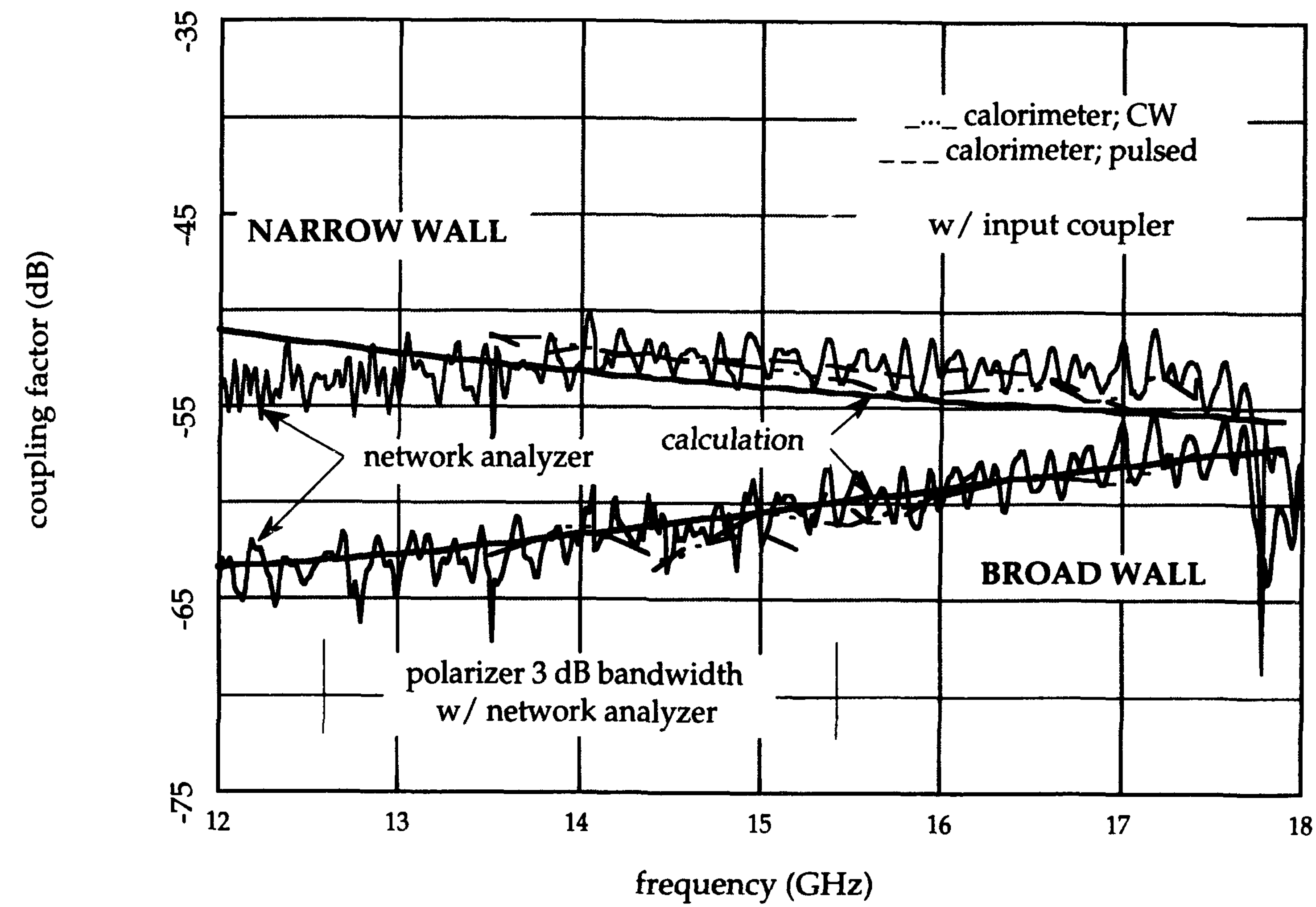


Figure 2.41. Comparison of output coupler coupling factor; network analyzer and calorimeter measurements.

2.7 CALORIMETER/WATER LOAD.

2.7.1 Introduction.

The ubitron is terminated with a combination water load and water flow calorimeter. This choice is dictated by two major considerations, high power capability and good broadband microwave matching characteristics to minimize oscillations. The ubitron must be terminated with a high-power load that will accommodate a peak microwave power of 1-5 MW and an average power of approximately 1-kW at the design operational parameters. Also, it must provide a broadband match to linear or circular polarizations and the TE_{11}, TE_{21}, and TM_{01} modes. These considerations are satisfied with a combination water flow calorimeter/load constructed from a series of coaxial nested cones.

In addition to terminating the ubitron microwave circuit, the water flow load is also used as a calorimeter. Calorimetry is the most fundamental of methods used for power measurement, and is, therefore, the reference power diagnostic for the ubitron. Peak power, in this case, is determined from the microwave pulse shape and measurements of the temperature rise in the water load caused by the absorption of microwave energy.

2.7.2 General Design Considerations.

In a fluid flow calorimeter, power is determined from two measurements, the fluid temperature rise caused by the absorption of energy and the fluid flow rate. The power response of an ideal fluid flow calorimeter is given by the following equation: $P = F(C_h D)T$, where

F = flow rate (cm^3/sec)
C_h = fluid specific heat (J/g°C)
D = fluid density (g/cm^3)
T = inlet-outlet temperature difference (°C).

The working fluid for the ubitron calorimeter is deionized water, so that C_h = 4.1796 J/g°C, and $D = 0.997$ g/cm^3 [19].

The calorimeter/load design is subject to three potentially conflicting requirements, good microwave matching characteristics, short response time, and reasonably large transducer output. First and foremost, the fluid volume must be sufficiently large to insure complete absorption of the incident RF. This must be matched against the need for a sufficiently short response time and, therefore a small volume, to allow rapid acquisition of data. The calorimeter response time, or time required to reach thermal equilibrium after the application of power, can be defined as $t_f = V/F$, where V is the enclosed fluid volume and F is the flow rate. A short response time would therefore require a small volume and/or a high flow rate. A high flow rate would, however, reduce the temperature rise of the fluid, and, therefore, the signal amplitude from the temperature transducer.

Determination of the appropriate water channel thickness begins with the permittivity of water, since microwave power is absorbed in water through dielectric effects. Assuming a complex permittivity, $\varepsilon = \varepsilon' - j\varepsilon''$, the absorption coefficient is [20]:

$$\alpha = \frac{\omega\sqrt{\mu\varepsilon'}}{\sqrt{2}}\left[\sqrt{1 + \left(\frac{\varepsilon''}{\varepsilon'}\right)^2} - 1\right]^{1/2} \quad (m^{-1}).$$

The penetration distance or e-folding length, $\delta = 1/\alpha$, is defined to be the distance at which the power is 1/e of the incident power. A plot of δ vs. frequency for water is shown in Fig. 2.42. This is calculated from permittivity data tabulated in Ref. 21. The maximum δ over the 12.4 to 18 GHz frequency range is approximately 2-mm. Neglecting the effects of the quartz cones and assuming normal incidence, a water channel thickness of 5δ (~1 cm) would absorb ~ 99.3% of the incident power. This would constitute a very good microwave match.

With this water channel thickness and other dimensions detailed below, the calorimeter response is a reasonable compromise between response time, microwave match, sensitivity and power handling capability. Assuming a minimum measurable temperature differential of 0.1°C, the theoretical calorimeter sensitivity is approximately 160 mW average or 1.1 kW peak (2.4 ms FWHM, 10^{-4} D.F.), at the lowest flow rate. However, a measurement would require approximately 17 minutes at this rate. This is reduced to a more

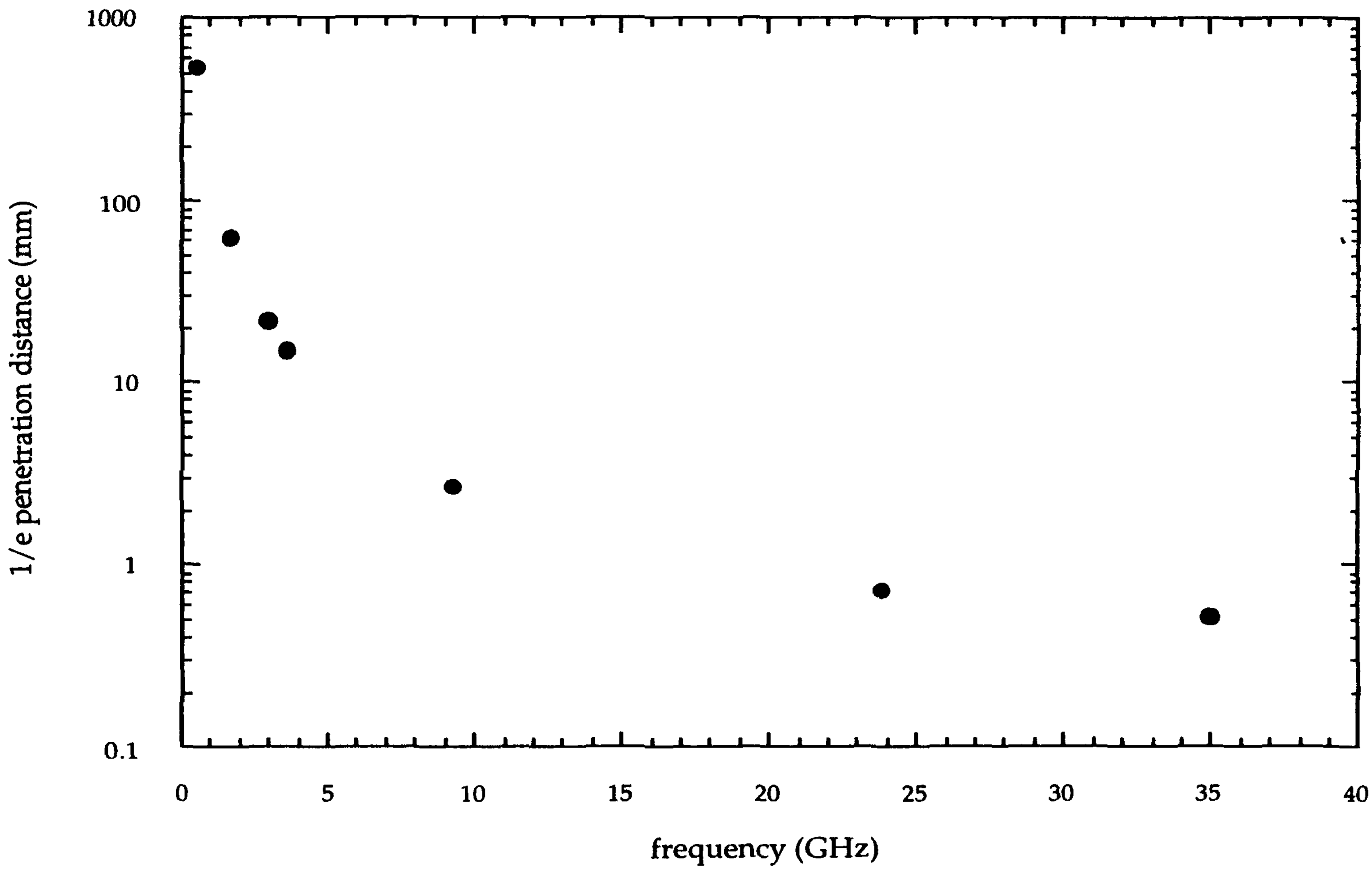


Figure 2.42. Penetration depth versus frequency for water.

reasonable 2 minutes at 200 ml/min. A 5 MW peak power pulse at the same duty factor and at the highest flow rate would raise the water temperature ~5°. If greater flexibility in absorption characteristics is required, the water channel thickness and/or radial profile can be modified without necessitating disassembly of the ubitron vacuum envelope.

A final general consideration concerns errors. The primary errors arise from inaccurate measurements of flow rate and temperature difference. However, an additional error term arises from fluid heat loss to the surroundings. This term can be expressed as T/2R, where T/2 is the average temperature rise of the fluid, and R is the thermal resistance to the surroundings. If the thermal resistance is high, this term can be neglected. Note that this term is proportional to the fluid temperature rise. Therefore, the calorimeter system must balance this error term for maximum sensitivity with minimum heat loss. Additional errors can be introduced by nonuniform flow rates and air bubbles in the system.

2.7.3 Configuration.

2.7.3.1 Mechanical Design.

The calorimeter is constructed from a series of three coaxial nested cones, with a common half-angle of approximately 6° and apex facing away from the incident radiation. An engineering drawing, to scale, is shown in Fig. 2.43. This configuration was chosen to distribute microwave energy deposition in the water as well as to present a good impedance match. The length of the inner cone, from maximum radius to apex, is approximately 10 axial wavelengths. Two quartz cones, each approximately 1/16 in. thick and separated by an air gap, were chosen to protect the electron gun from a possible water leak. The innermost cone forms the vacuum envelope, and the outer quartz cone forms the inner boundary of the water channel. The outermost cone is a 10 mil polycarbonate sheet, chosen to minimize conduction losses. Thermal isolation is provided by a stagnant air gap between the main calorimeter assembly and the polystyrene foam insulated aluminum outer case.

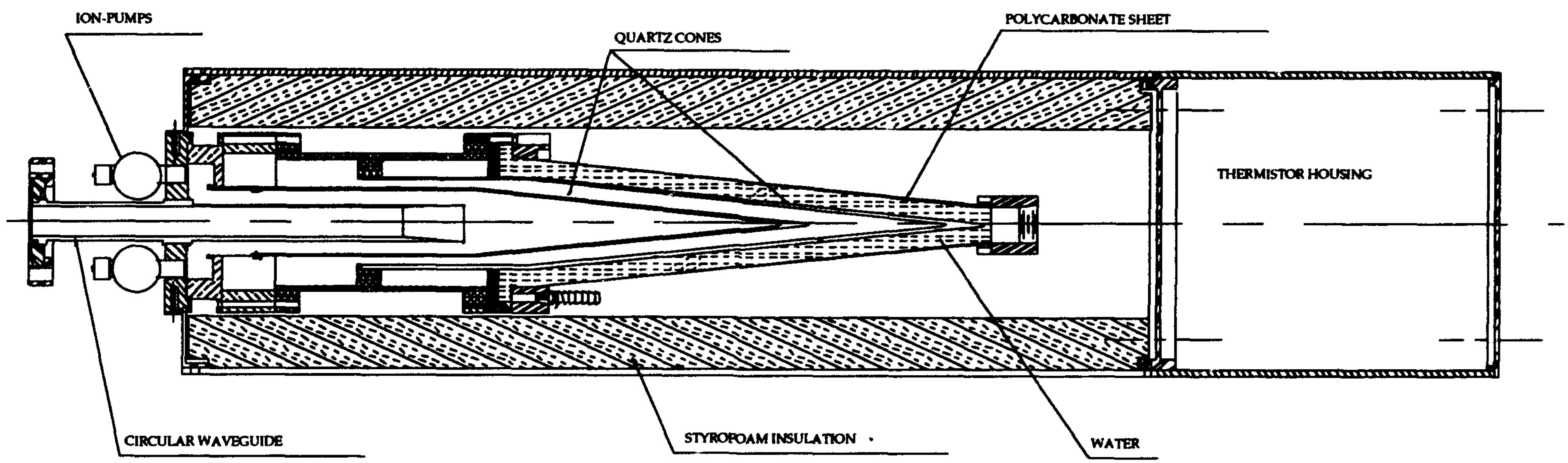


Figure 2.43. Calorimeter/water load configuration.

Deionized water is pumped into a single port at the cone apex by a 23-2300 ml/min peristaltic pump with flow integrator to four ports at the large radius end. The flow rate is monitored with a digital meter accurate to 1%. Separate linearized thermistors are used to measure the inlet and outlet water temperatures. The thermistors are placed directly in the input and output water streams and are electrostatically isolated from possible direct microwave pickup. An assembly drawing for the thermistor assembly is shown in Fig. 2.44.

A removable resistance heater for calibration can be inserted from the apex end. A block diagram for the complete calorimeter system is shown in Fig. 2.45.

2.7.3.2 Electrical Design.

Linearized thermistors were chosen as the temperature transducer for two reasons: high sensitivity and simplicity; basic operation does not require conditioning circuitry when used in the resistance mode. However, additional flexibility results from a possible voltage mode of operation. This allows easy connection to data acquisition systems. The thermistors were manufactured by Yellow Springs Instrument Co., Models YSI 44201 and 44202. With the appropriate linearization resistors, the thermistor composite resistance is linearly proportional to temperature. Each model's accuracy and interchangeability is specified as ± 0.15°C over the -30 to 100°C temperature range. The thermistors can be used individually or connected for a direct differential output when in the voltage mode.

2.7.4 Calibration.

Before checking the entire calorimeter system performance, the major subsystem components, thermistors and flow meter were checked.

2.7.4.1 Flow Meter.

The digital flow meter accuracy is specified as ± 1% full scale (1999 ml/min). However, when checked against the time to fill a known volume, the meter reading was found to be approx. 3.5% low. All flow meter readings in calibration and normal use have been corrected.

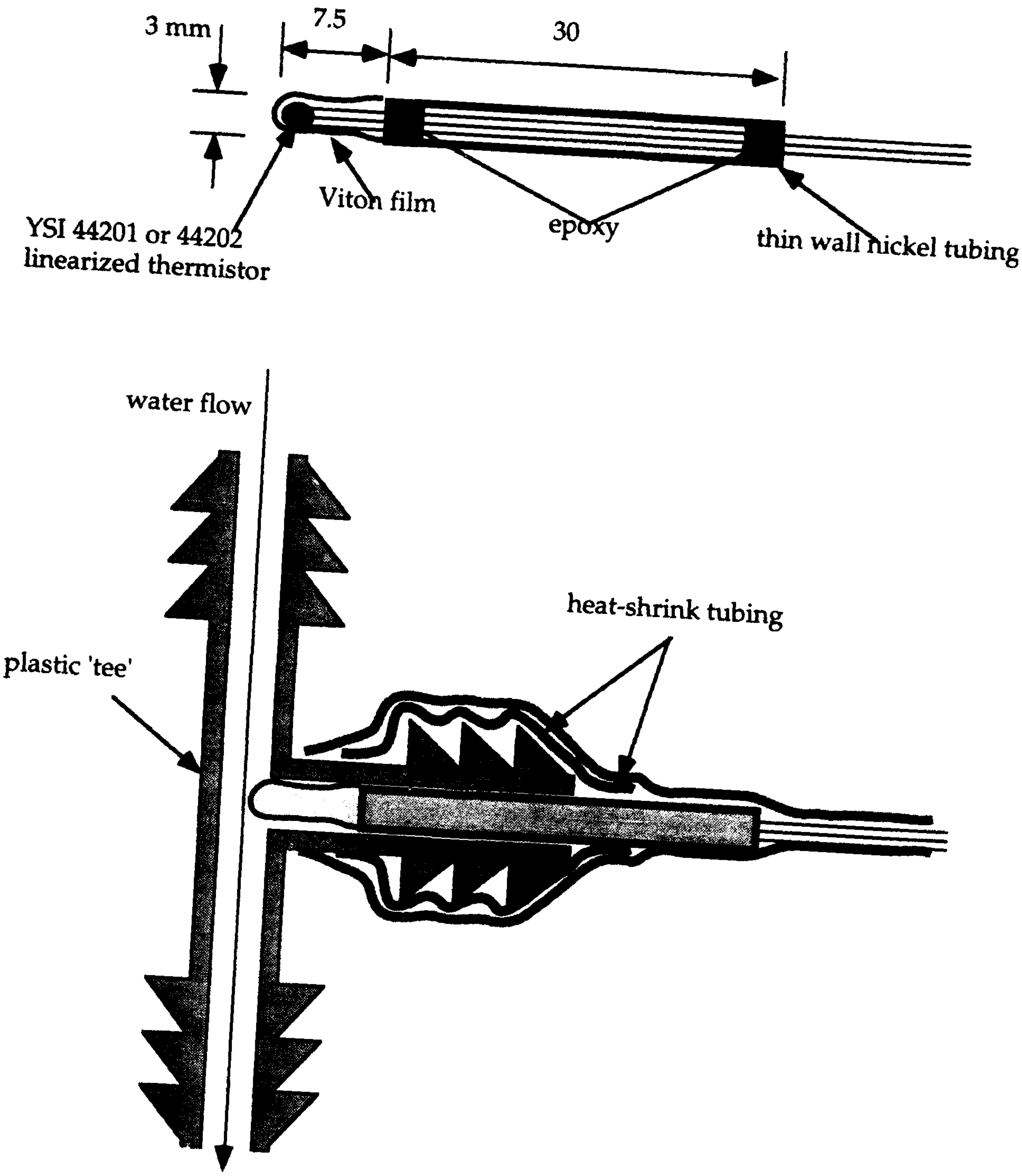


Figure 2.44. Thermistor assembly.

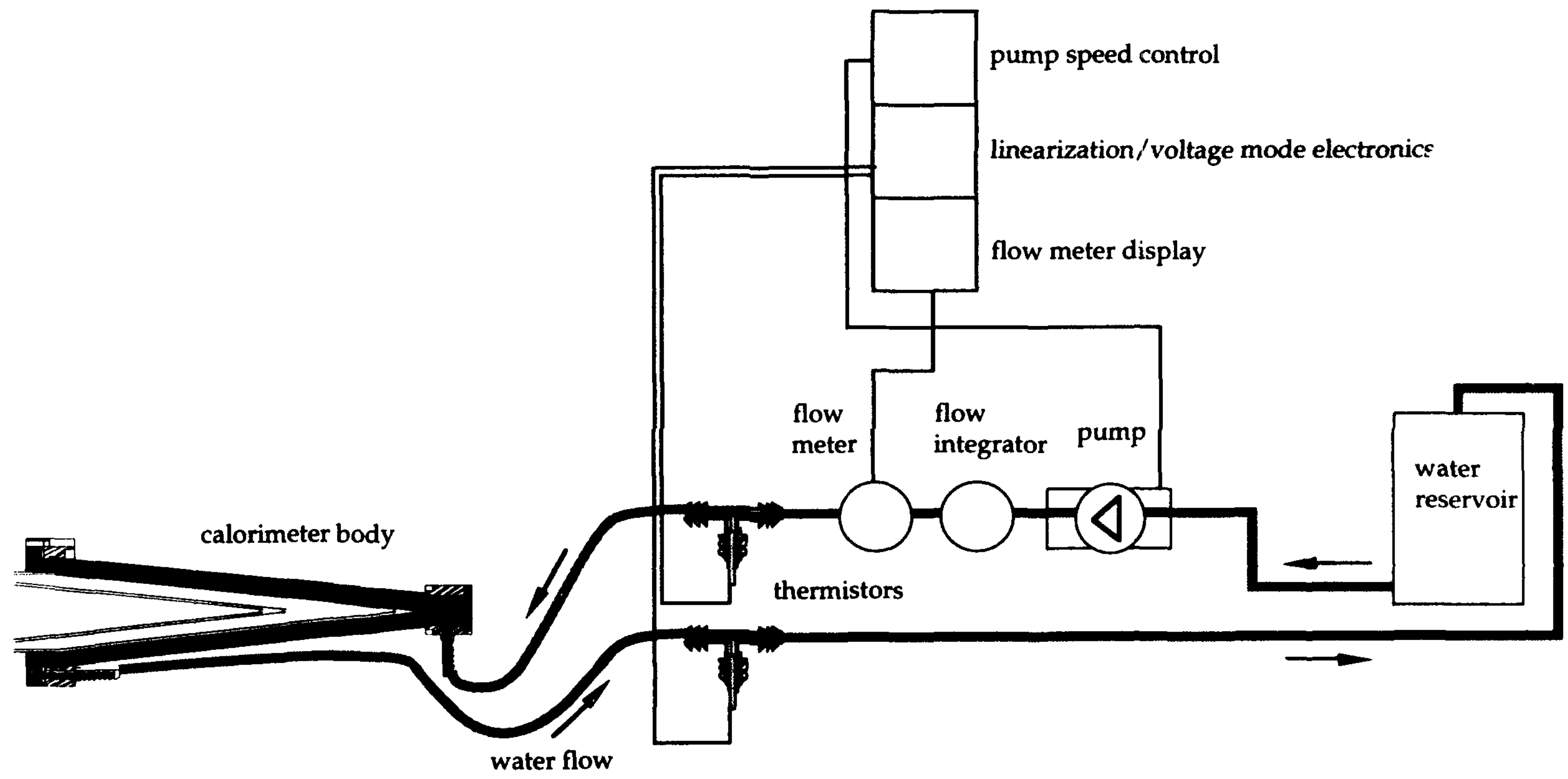


Figure 2.45. Calorimeter system block diagram.

2.7.4.2 Thermistors.

Depending on thermistor model, the nominal resistance vs. temperature equation is specified over a 0 - 100°C (44201) or a -5 - 45°C (44202) temperature range. Either temperature range is larger than expected under normal experimental conditions. The thermistors were, therefore, calibrated over a more restrictive 20-30°C temperature range by comparison with a 0.1°C resolution, mercury thermometer with calibration traceable to the National Bureau of Standards. Each temperature transducer, thermistor and thermometer, was connected to essentially the same point in the pumping system. Water temperature was varied by adding hot water to the reservoir, and each transducers' response was recorded. Typical thermistor temporal response curves are shown in Fig. 2.46. The resulting calibration equations are: R_4 (input) = 2759.2 - 17.0912T(°C) and R_5 (output) = 2752.9 - 16.808T, compared to the nominal R = 2768.23 - 17.115T (0 - 100°C).

2.7.4.3 Resistance Heater Calibration.

The basic calibration of calorimeter system accuracy was performed using a resistance heater connected to a DC power supply and inserted through the apex end of the calorimeter. The system response was determined by calculating the applied power based on the measured differential temperature rise and flow rate, using the ideal calorimeter power response equation and the individual thermistor calibrations. To allow for a possible thermistor temperature differential with no power applied, possibly due to different thermal equilibration times for each thermistor, the total temperature differential with power applied is calculated as follows:

$$\Delta T = \left(\frac{R_{f5} - R_{i5}}{b_5}\right) - \left(\frac{R_{f4} - R_{i4}}{b_4}\right),$$

where b_4 = -17.0912, b_5 = -16.808, and subscripts i and f refer to the initial and final resistances, respectively. A typical calorimeter response graph is shown in Fig. 2.47 for a flow rate of 841 ml/min and applied power ranging from 9.8 to 158 W. The final calibration graph, for a variety of flow rates and powers, is shown in

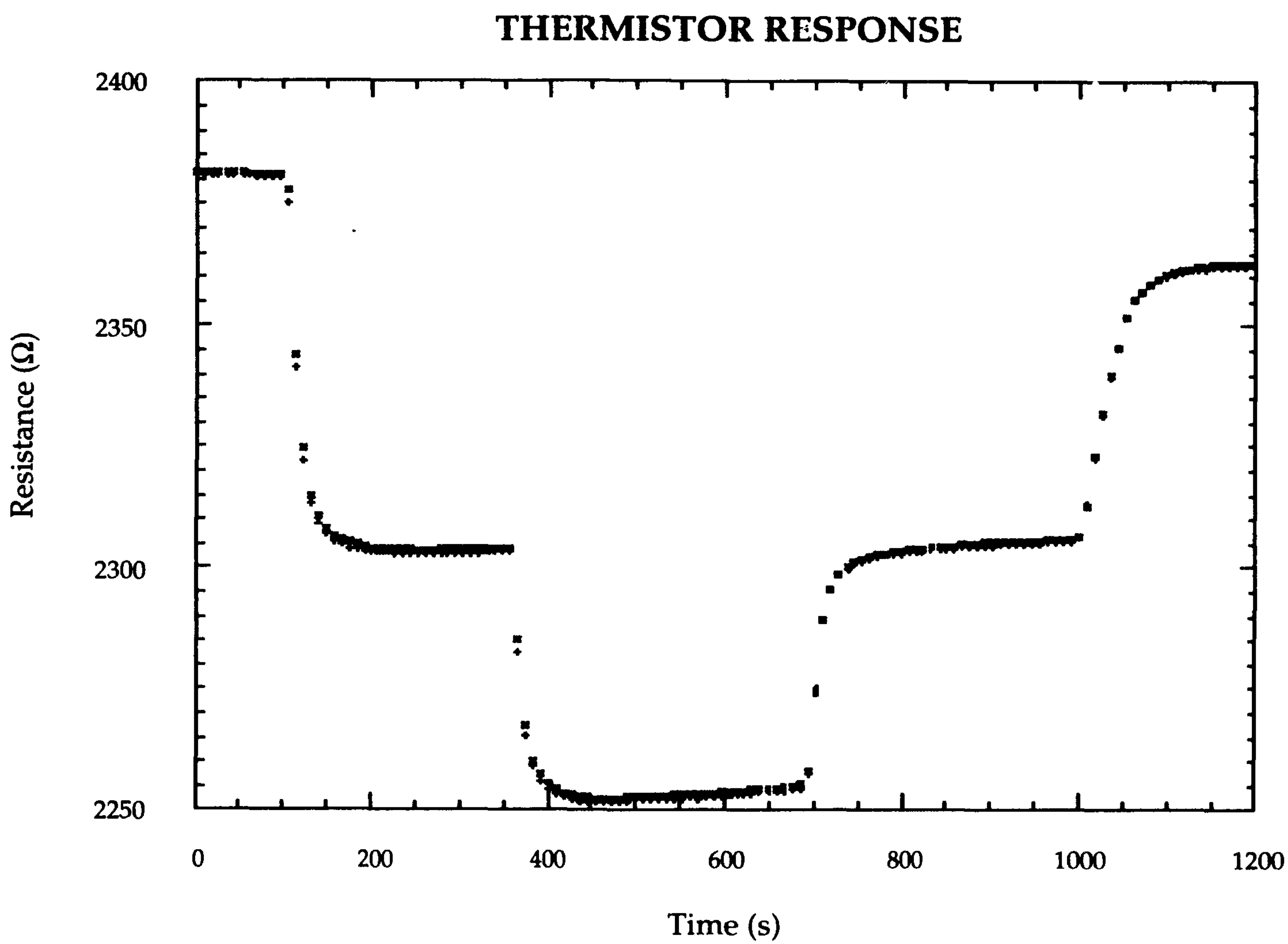


Figure 2.46. Typical thermistor temporal response curves.

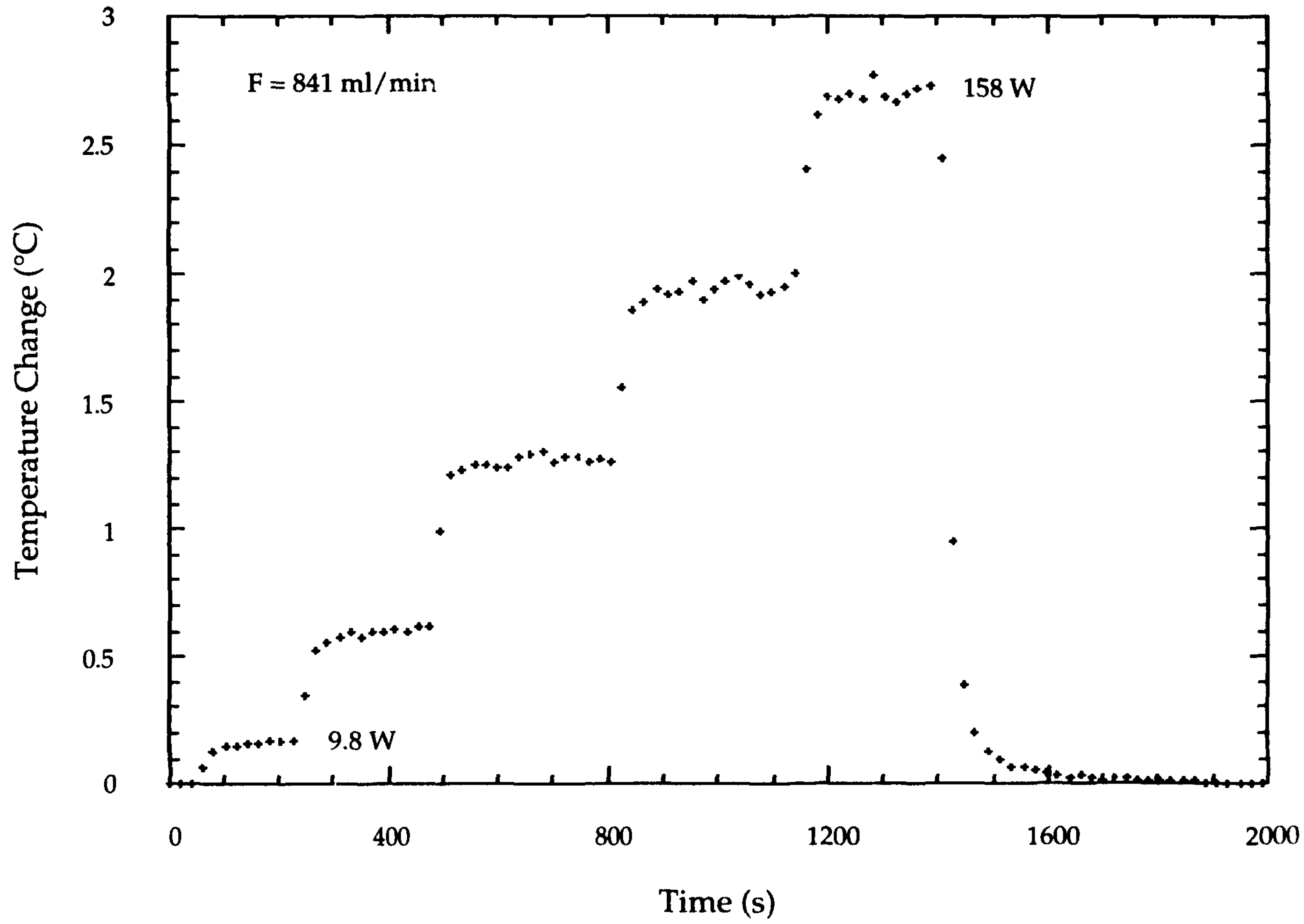


Figure 2.47. Typical calorimeter temporal response graph.

Fig. 2.48. The agreement is quite good, the calculated value being within ~ 1.6% of the applied power.

2.7.5 Microwave Performance.

In addition to measuring the applied power accurately, the calorimeter/ load must also present a good match to the microwave circuit. A good match in this case is defined as a return loss of 20 dB or better, which represents reflection of less than 1% of the incident power. The chosen quartz cone configuration does result in such a match [see Fig 2.43.] Fig. 2.49 shows the return loss measured for the two-cone system when immersed in a large dewar of water. For this measure-ment, the microwave launching horn was inserted to the beginning of the taper section of the inner quartz cone, with the water level 0.5" above the horn end. The inner and outer cone tip separation was 1.6". This is for linear polarization.

When the entire calorimeter system was assembled in its final configuration, the return loss deteriorated somewhat (Fig. 2.49). The return loss reduction at low frequencies is not significant. However, the return loss reduction for frequencies above ~ 17.4 GHz is possibly a problem, since the cutoff frequency for the TE_{21} mode is 17.8 GHz. This problem has been traced to the Conflat flange/gasket arrangement used to connect sections of the ubitron together.

As a final test of calorimeter performance, the response of the calorimeter system to pulsed microwave power was measured (Fig. 2.50). For this test, the linearly polarized incident microwave radiation was generated by the high power amplifier, operated at a duty factor of .022 to .0589 in order to generate sufficient average power for thermistor measurements. Measurements were taken at 14.5 and 15.9 GHz at several flow rates and power levels. Microwave power was measured with a calibrated 50-dB coupler and a Wavetek Model 8502 peak power meter with detector.

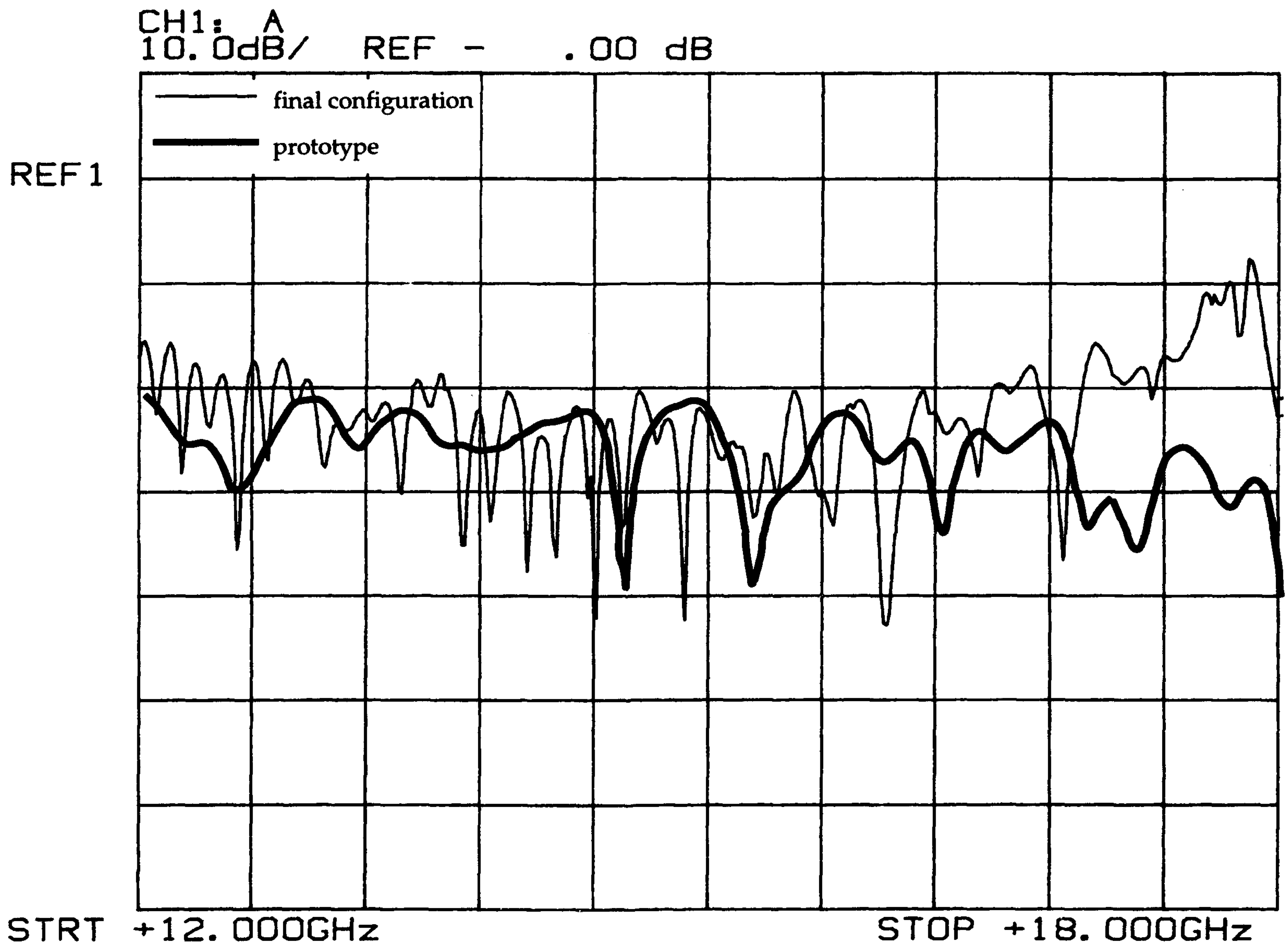


Figure 2.49. Return loss of two-cone calorimeter configuration.

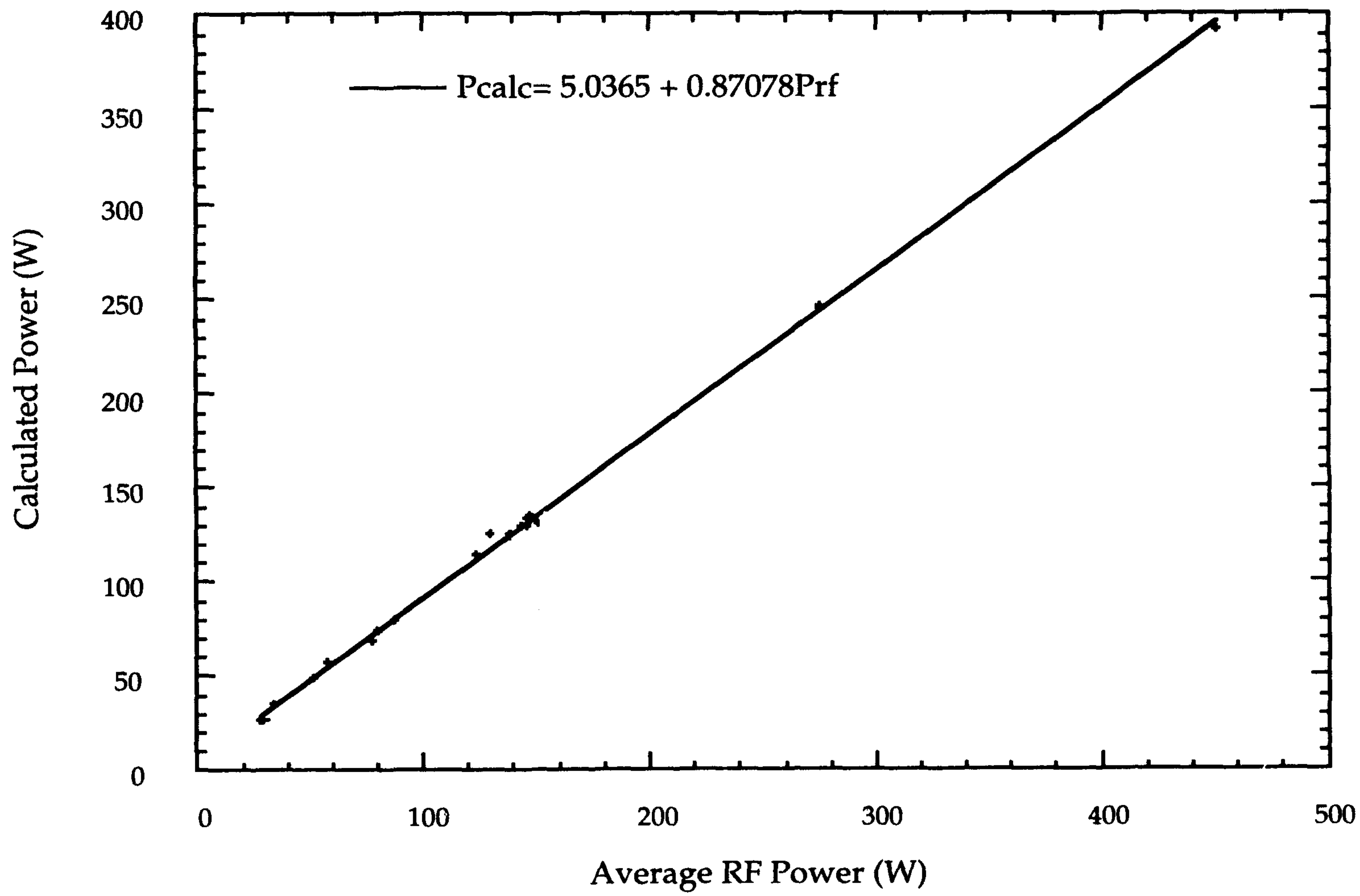


Figure 2.50. Calorimeter calibration using pulsed microwaves.

SECTION 3
PERFORMANCE

3.1 RIPPLED BEAM PERFORMANCE.

3.1.1 Introduction.

During the initial tests of the NRL ubitron amplifier tube, the tube was operated below design voltage and with a highly scalloped electron beam in order to investigate harmonic interactions. Both amplification and oscillation were measured in K_u-band, depending on operational parameters. All interactions were harmonic since the fundamentals of the cyclotron maser, ubitron, and periodic position mechanisms were below cutoff of the TE_{11} waveguide mode.

The performance characteristics discussed below were measured using the modified SLAC klystron gun. As part of the gun conditioning procedure, current was first drawn from the gun at voltages much lower than 250-kV, starting around 30-40 kV. The operating voltage was then slowly increased until arcing occurred, at which time the voltage was slightly reduced below the arcing threshold. After stable operation at this voltage for a period of time, the voltage was again slowly increased until arcing occurred. Weeks of this procedure were required to approach the nominal 250-kV operating voltage. During this conditioning period several interesting amplification and oscillation regimes were discovered when the gun trim coil current was reversed, effectively cancelling the axial magnetic field at the cathode surface.

SCRIBE simulations show that subsequent strong over-focusing with the solenoidal field results in a solid beam with a high degree of scalloping at the edge and trajectories which focus close to the axis. Perpendicular to parallel velocity ratios (alpha) of approximately 0.4 or greater can easily be generated in this manner, and are adjustable with a trim coil. (See Fig. 2.3 in Section 2.1.2. for a comparison of rippled and laminar beam trajectories.) This system provides a simple and flexible means of generating a beam with either high or low velocity ratio depending on trim coil operation and gun position relative to the solenoid.

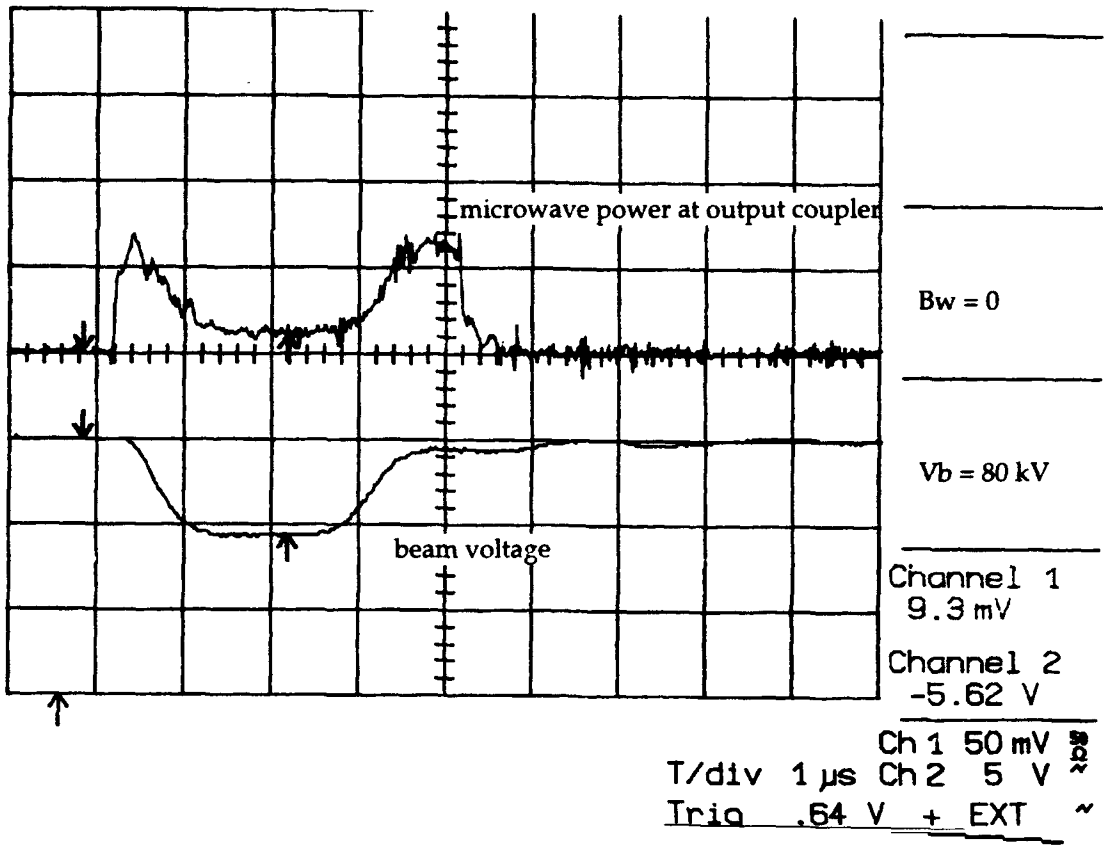
Refer to Section 1 for a discussion of the experimental configuration, including tube components, microwave circuits, diagnostics, and major subsystems. No analysis of results will be presented, since no interaction model was readily available. See, however, Ref. 22 for relevant discussions.

3.1.2 Amplifier Results.

Amplifier operation in mid-K_u-band was observed at a beam voltage of ≈70-80 kV and an axial magnetic field of ≈2.9 kG with an unoptimized bandwidth of at least 7%. A complete characterization of amplifier operation was not performed due to limited time. The data presented below are, however, indicative of an interesting parameter regime for amplifier operation. It should be noted that some measurements identified as 'amplifier' are not conclusive, since both input and output frequencies were not measured and some wave shapes were somewhat unstable. However, the preponderance of measurements exhibit characteristics consistent with amplifier operation.

The first measurements demonstrating gain are shown in Fig. 3.1 for the following parameters: V = 80 kV, I = 15 A, and B_z = 2.4 kG. Two sets of curves are shown; the upper set corresponds to finite P_{in} and B_W = 0, and for the lower set B_W = 175 G. The upper trace in each set is the detected microwave power transmitted through the ubitron, and the lower trace is the beam voltage. Approximately 3 dB gain is observed. An important feature to be noted is the reduction in microwave power injected into the tube during the beam pulse caused by a current-dependent mismatch at the input coupler. This is shown more explicitly in Fig. 3.2, where the uppermost trace is the transmitted microwave signal without the beam, and the middle trace is the transmitted signal with the beam for the voltage pulse shown in the lower trace. The wiggler is off in both cases. To account for beam loading effects in all reported gain calculations, P_{in} is the transmitted power with the beam on and wiggler field off, and P_{out} is the transmitted power with both beam and wiggler on.

Typical measurements indicating amplifier operation are shown in Fig. 3.3 for the following parameters: V = 71 kV, I = 12 A, B_z = 2.8 kG, B_W = 175 G, f = 14.4 GHz, and P = 2.6 kW (at phase splitters). In this case, the upper set corresponds to finite wiggler field, but P_{in} = 0, and the lower set corresponds to



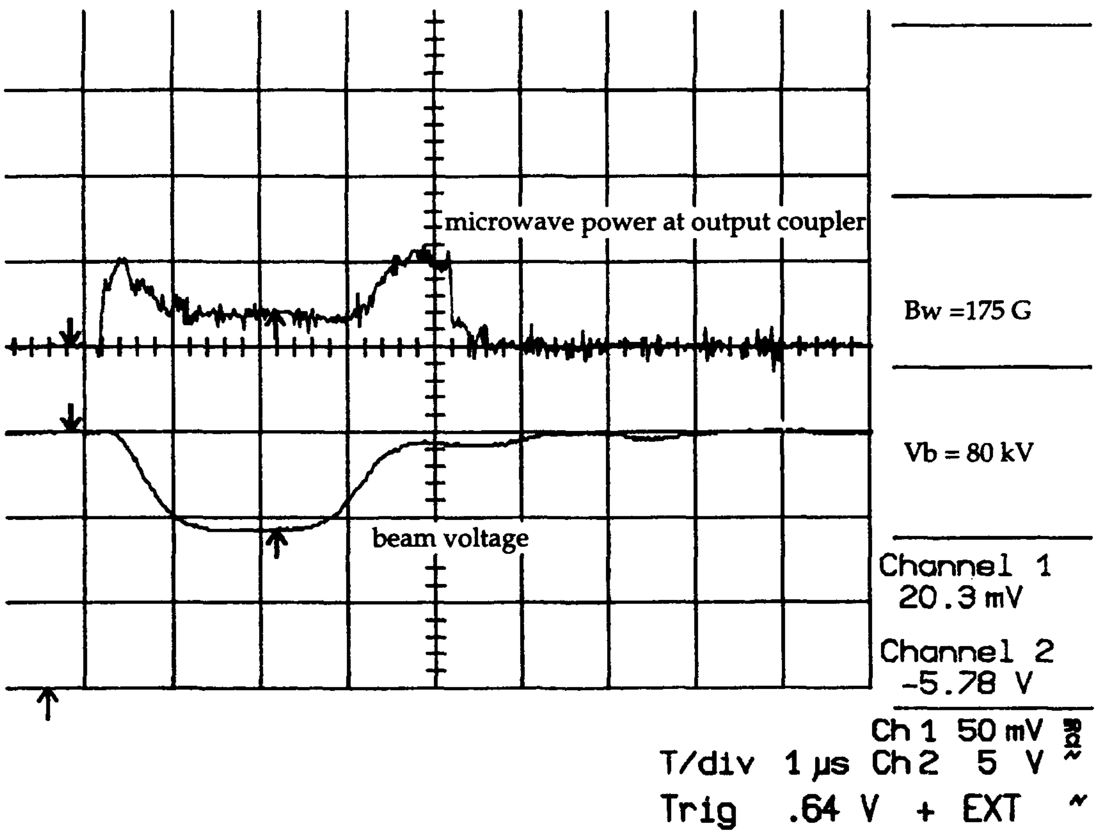


Figure 3.1. First rippled beam gain measurements: V=80kV, 1=15A, Bz=2.4kG.

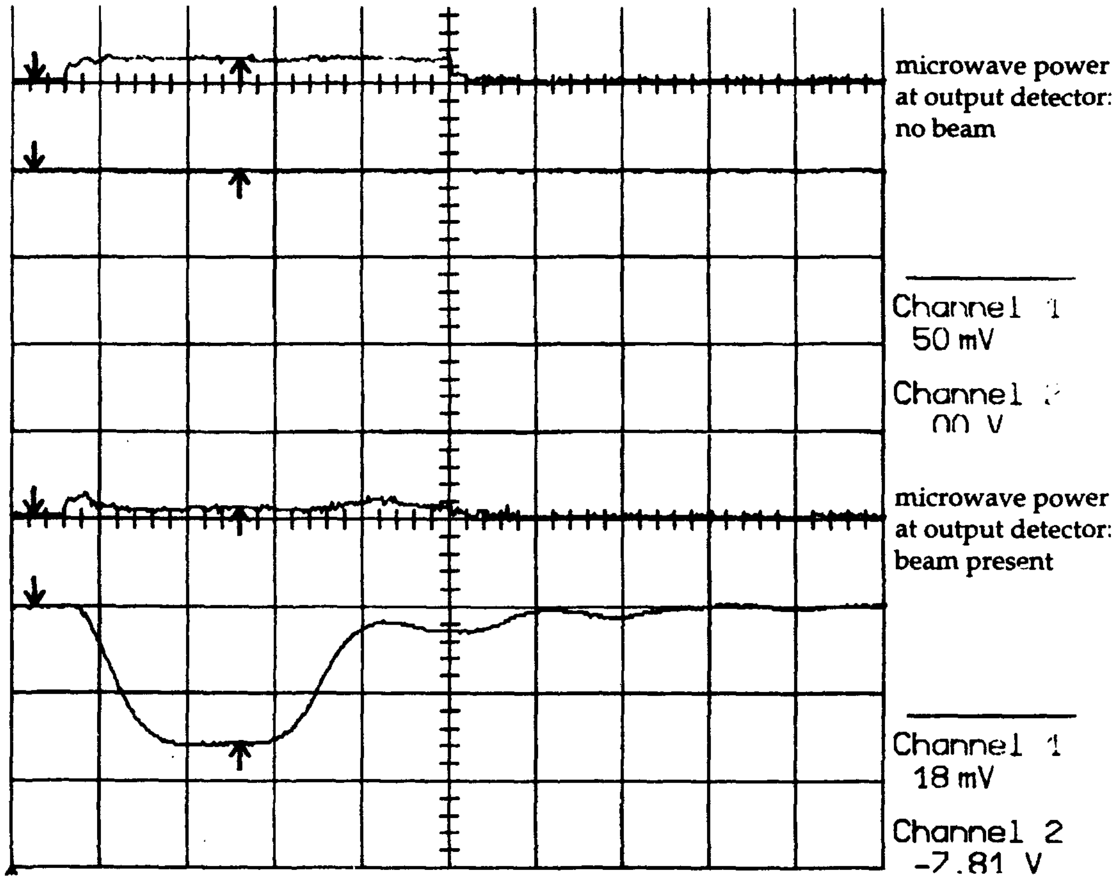


Figure 3.2. Demonstration of beam current effect on input coupling.

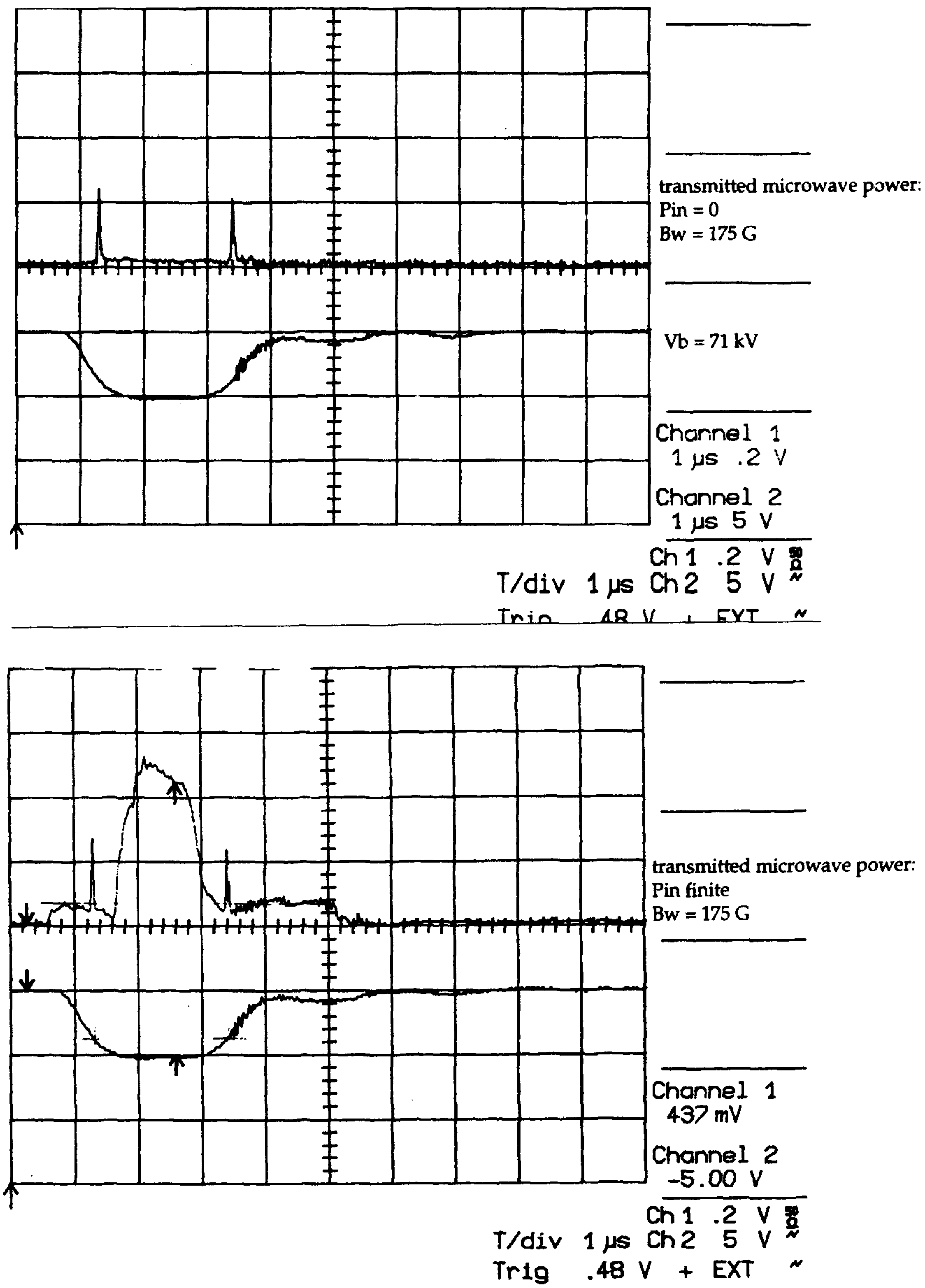


Figure 3.3. Rippled beam amplifier waveforms: V=71kV, I=12A, Bz=2.82kG, Bw=175G, f=14.4GHz, Pin=2.6kW@phase splitters.

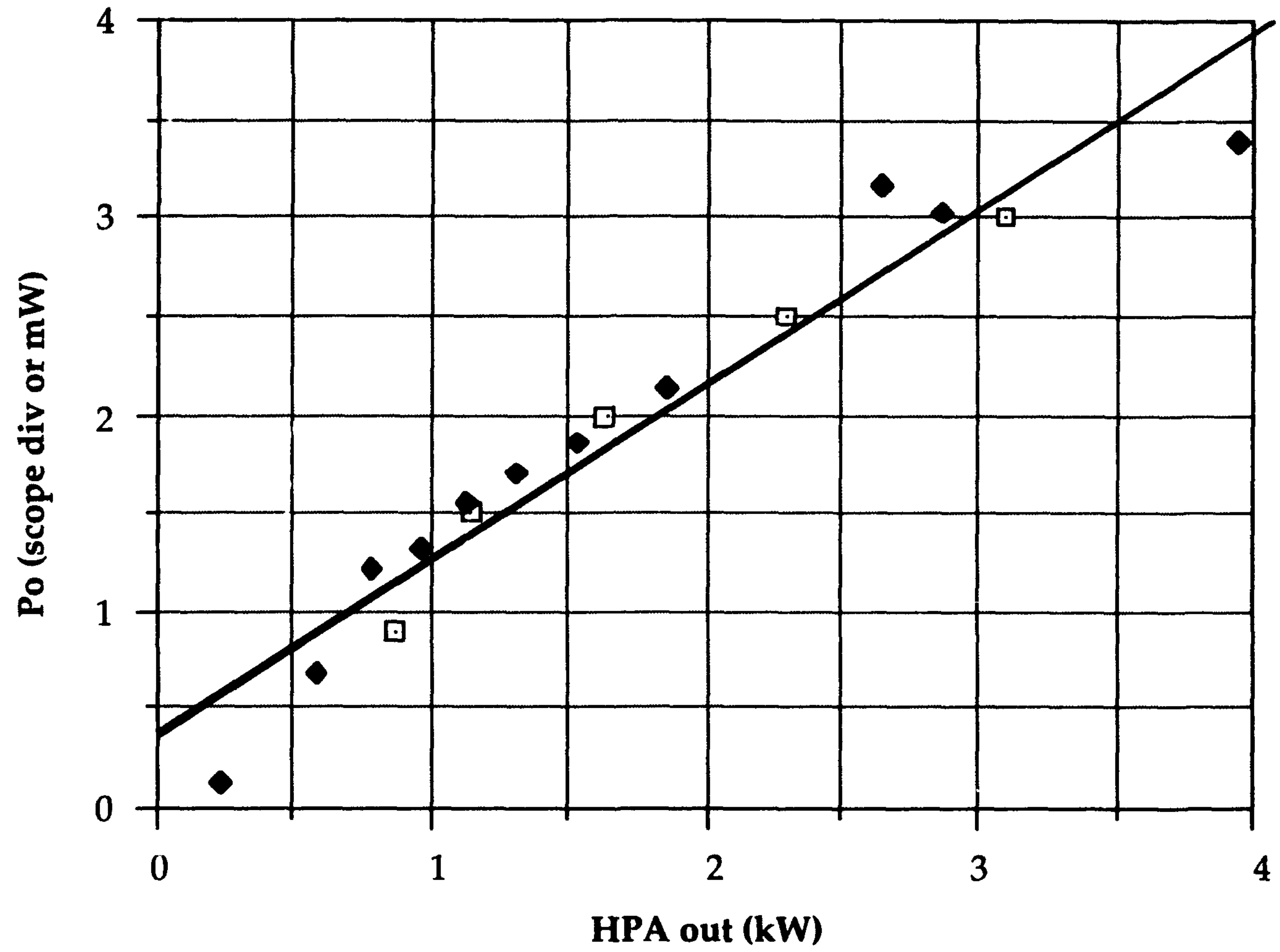
finite wiggler field and finite P_{in}. Brief periods of oscillation are observed on the leading and trailing edges of the beam pulse, indicated by 'rabbit ears' in the microwave signal. Although the output frequency was not measured, the measured output power is linearly dependent on input power, as shown in Fig. 3.4, including measurements for a second parameter set at 135 kV. A linear dependence of output power on wiggler field strength in the 'amplifier' mode is shown in Fig. 3.5.

Approximately 8 - 14 dB of gain was measured over a 14.4 - 15.4 GHz frequency band at V = 71 kV, I = 12 A, B_z = 2.8 kG, and B_w = 175 G (Fig. 3.6). For verification, the measurements were repeated using approximately the same field values, but slightly lower V_b and I_t. This resulted in slightly higher gain values, ~19 - 24 dB, with a peak output power of 19 kW. Increasing the beam voltage and input power reduced gain slightly to 17 dB, while increasing output power to ~25 kW, measured calorimetrically. This corresponds to a 3% peak unsaturated efficiency. For both cases, the output frequency was measured and found to be the same as the input frequency, 14.7 GHz. A complete set of waveforms from a ~ 80 kV measurement series is shown in Fig. 3.7. Included are gun current and transmitted beam current waveforms. Amplifier operation was also measured at voltages as low as 40 kV.

As mentioned above, no theoretical analysis was performed. Comparison of the amplification frequencies with the uncoupled dispersion curves (Fig. 3.8) suggests the mechanism is either a periodic position or cyclotron maser interaction (both second harmonic) with the TM_{01} mode. Conventional cyclotron maser theory indicates low interaction with TM modes near cutoff, but due to the scalloped nature of the beam, periodic position interaction is possible. It should be noted that amplifier operation (and oscillation performance discussed below) are very sensitive to the trim coil current, which should not be too surprising given the over-focused nature of beam generation.

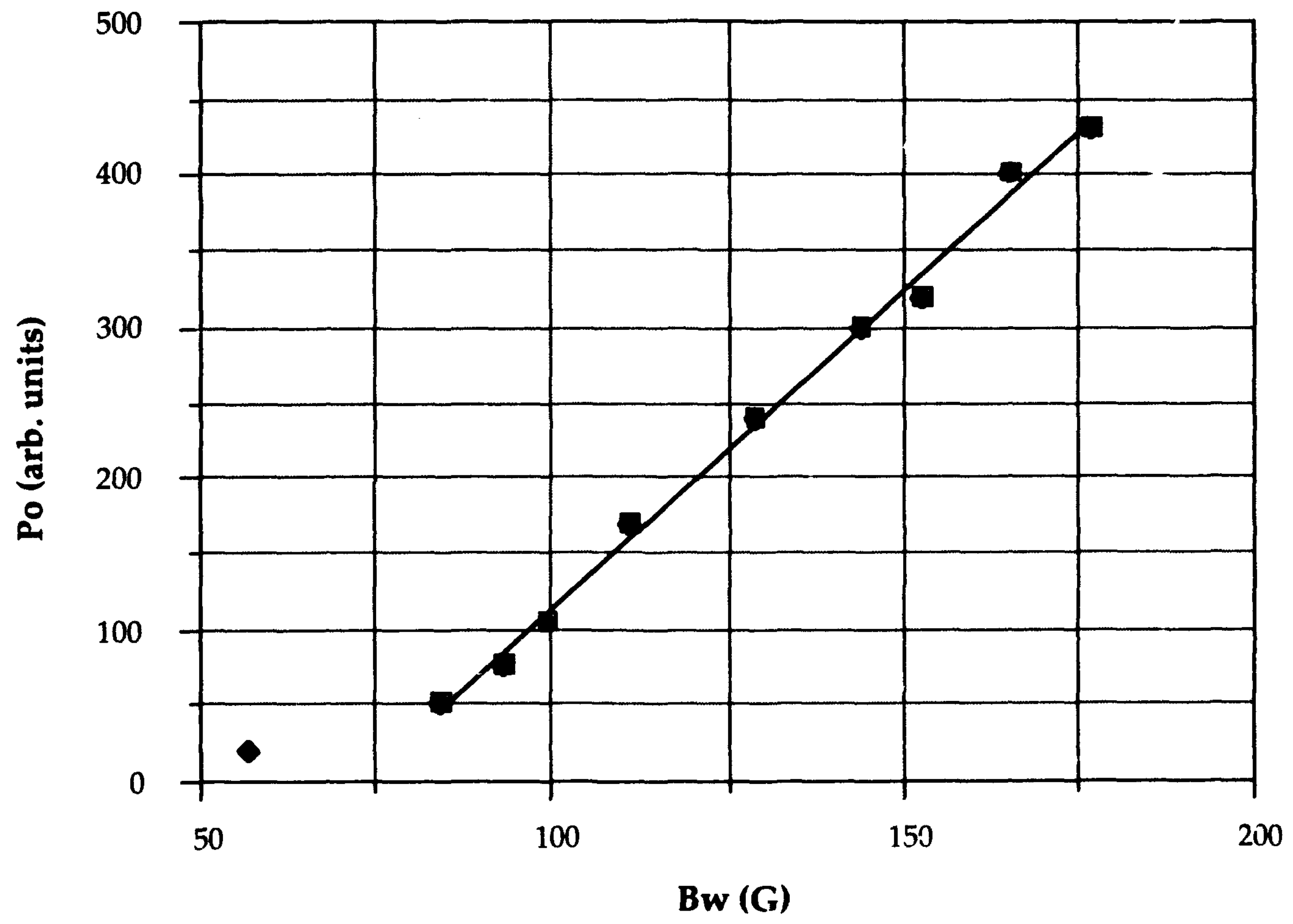
3.1.3 Oscillator Results.

As noted above, oscillation was also observed. While this typically took the form of narrow 'rabbit ears' on the leading and trailing edges of the beam pulse during amplifier operation, oscillation could also be induced without a



	Vbeam (kV)	Ibeam (A)	Bs (kG)	Bw (G)	freq. (GHz)
□	135	22	2.9	147	14.95
◆	71	12	2.8	174	14.37

Figure 3.4. Output power dependence on input power.



Vbeam (kV)	Ibeam (A)	Bs (kG)	freq. (GHz)
71	12	2.8	14.37

Figure 3.5. Output power dependence on wiggler field.

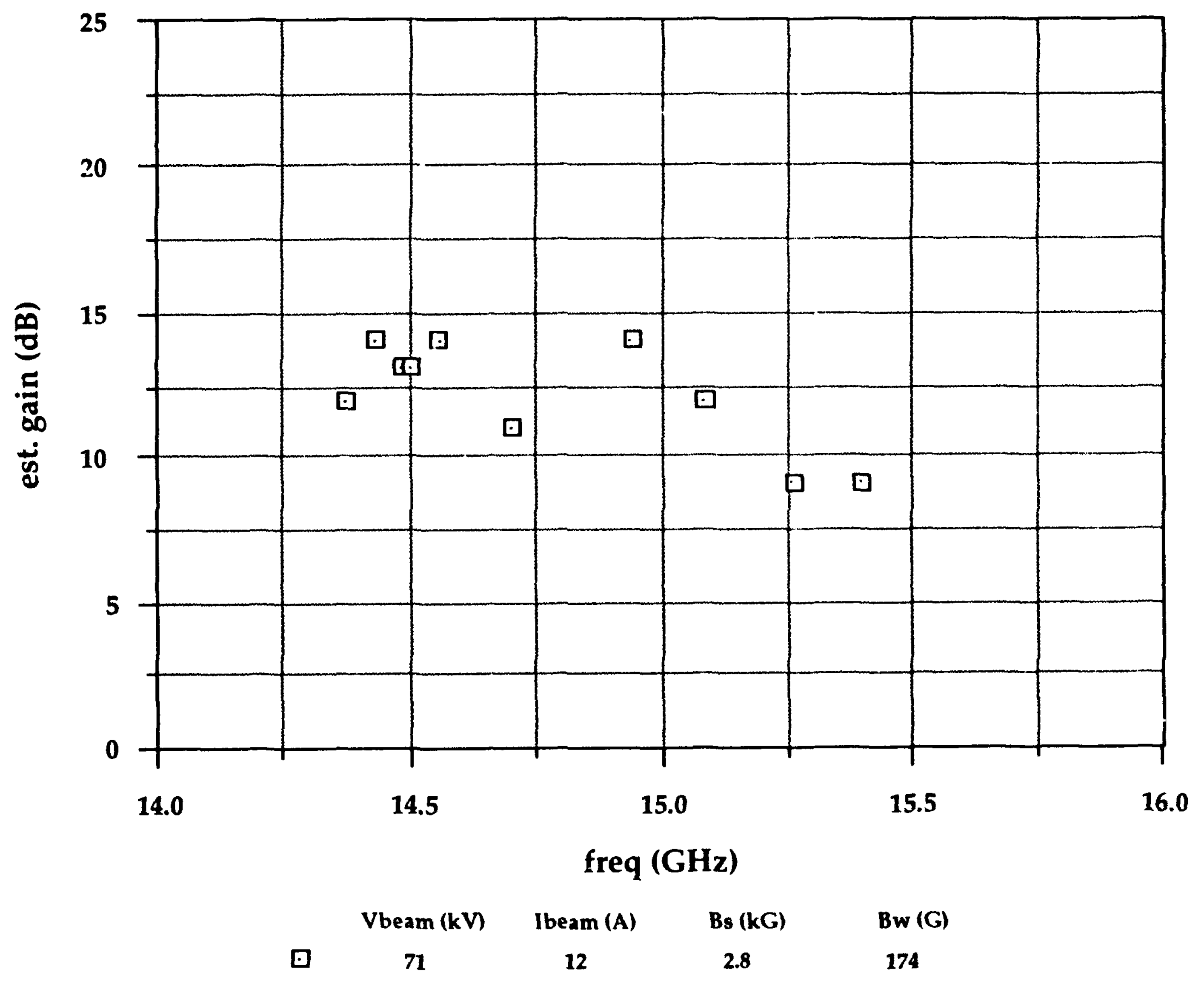


Figure 3.6. Gain vs. frequency: V=71kV, I=12A, Bz=2.8kG, Bw=175G.

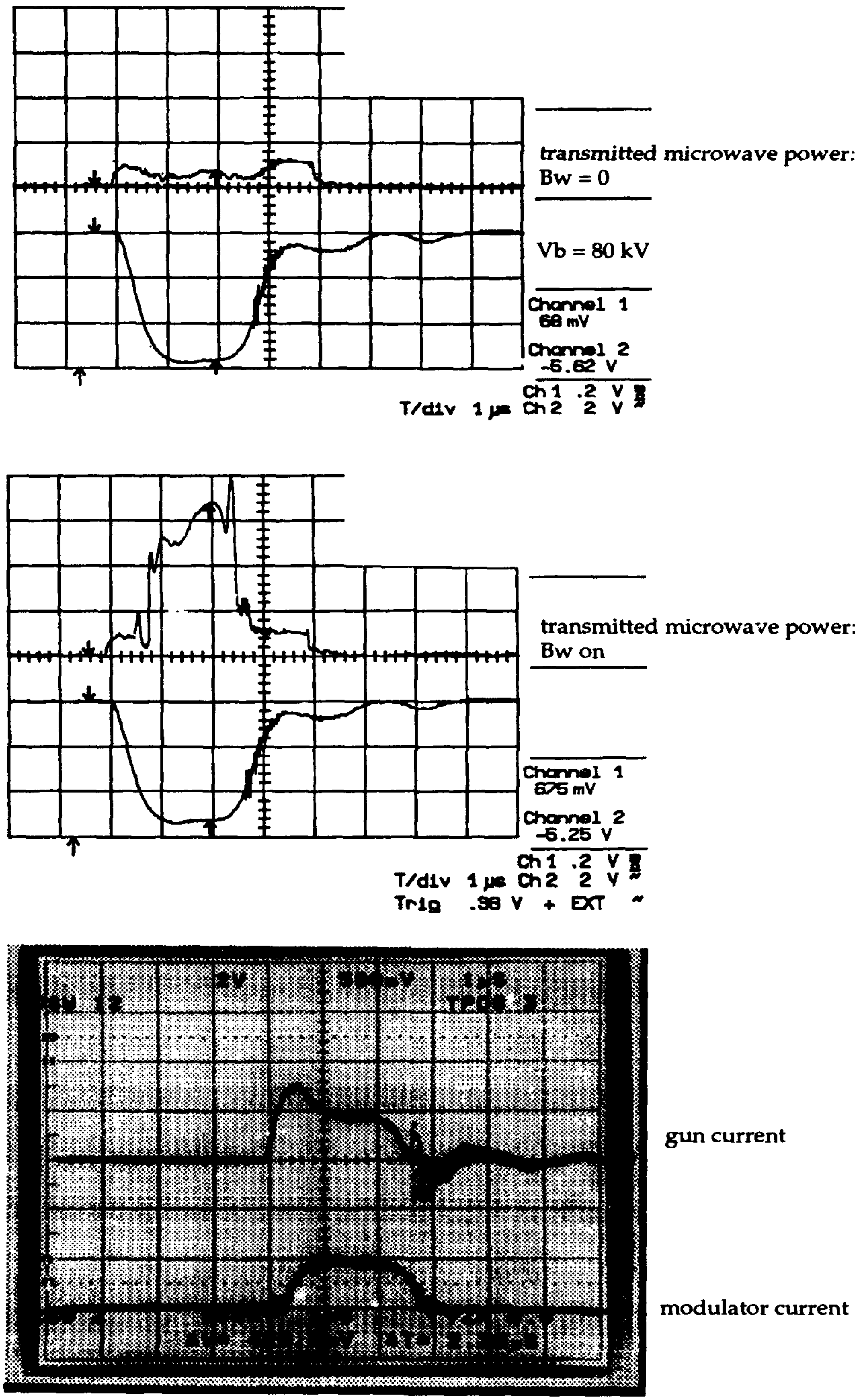


Figure 3.7. Typical waveforms: 80kV series.

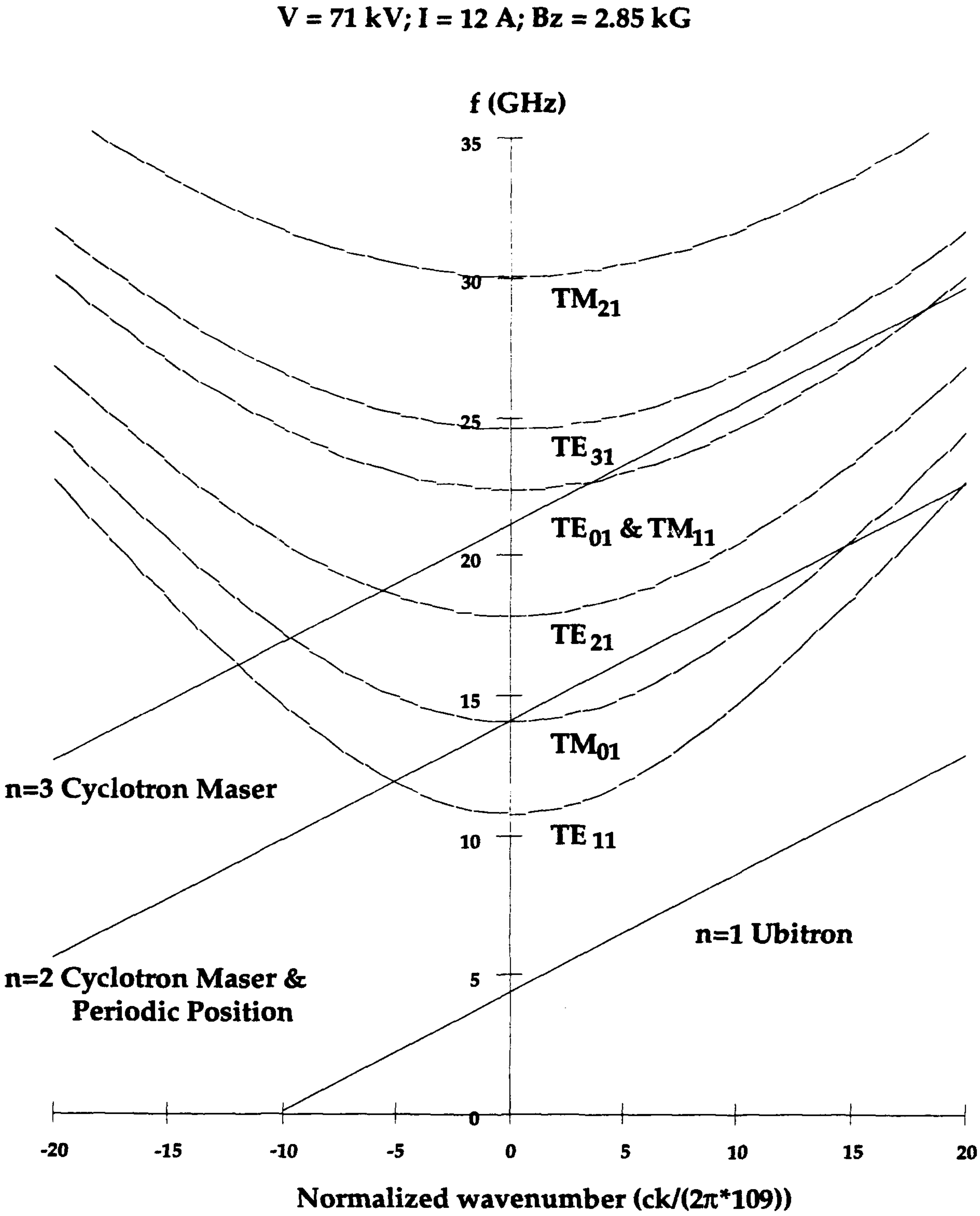


Figure 3.8. Uncoupled dispersion curves: V=71kV, I=12A, Bz=2.85kG, Bw=175G.

wiggler field for appropriate trim coil currents. Although oscillations typically grow from noise on the beam, it is possible that some of the measured oscillator signals were primed from the intermediate amplifier noise amplified by the HPA, without a drive signal from the sweeper.

Due to the narrow 'rabbit ear' pulse width and low repetition rate, it was not possible to measure the oscillation frequency directly. However, a frequency measurement could be made by reducing the beam modulator voltage to the oscillation onset voltage value of a higher voltage pulse, maximizing the microwave signal pulse width. Such a signal is shown in Fig. 3.9 for the following parameters: V = 27 kV, I = 4 A, B_Z = 2.9 kG, and B_W = 0 G. At this beam power level, it was possible to increase the modulator repetition rate sufficiently, for a frequency measurement to be made with an EIP 585 frequency counter, 17.9-17.99 GHz. This is very close to the cutoff frequency for the TE_{21} mode in an 8.15 mm radius cylindrical waveguide.

Further characterization of rippled beam oscillator performance was not attempted. However, it was observed that the application of microwave power could suppress the oscillation. An oscillator signal could be completely suppressed by injecting 4 kW (at phase splitters) at 14.8 GHz for the following parameters: V = 155 kV, I = 30 A, B_Z = 2.8 kG, and B_W = 85 G. This is demonstrated in Fig. 3.10. The uppermost trace is the detected output signal with applied power and the third trace is the oscillator signal in the absence of applied power. The bottom trace is the injected microwave signal from the HPA. This effect was also observed for injected power at 15.05, 15.2, 15.25, 15.37 GHz.

Again no theoretical analysis of rippled beam oscillation was attempted. Comparison of uncoupled dispersion curves suggests that the oscillation configuration (V=25 kV, I=3.5 A, B_Z=3.2 kG, f_{OSC}=17.9 GHz) [Fig. 3.11] is a second harmonic cyclotron maser interaction with the TE_{21} circular waveguide mode which occurs near the cutoff of the mode. The wiggler field is not necessary due to the high alpha generated by the electron gun and trim coil at low voltage. The exact nature of the effect of the wiggler field has not been determined, but the wiggler may act to enhance the beam alpha at higher voltages where the gun/trim coil do not generate as high an alpha beam.

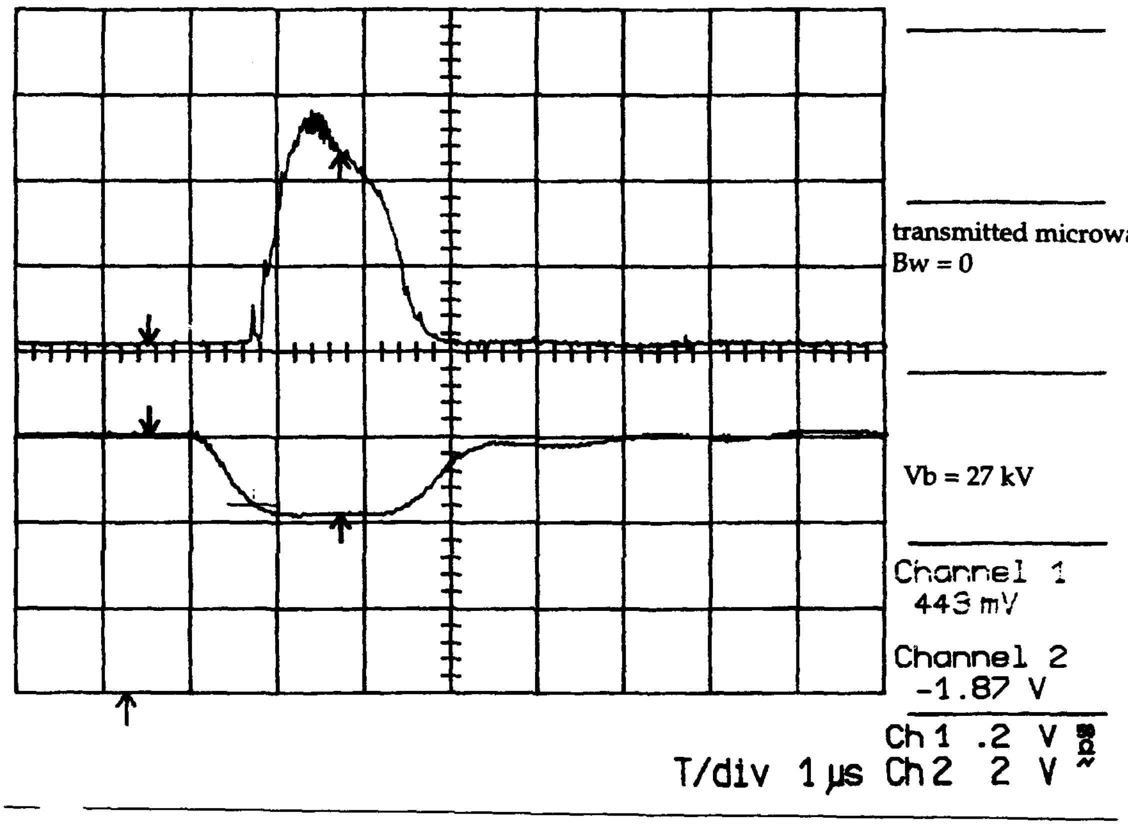


Figure 3.9. Oscillator waveform: Vf=27 kV, I=4 A, Bz=2.9 kG, Bw=0 G.

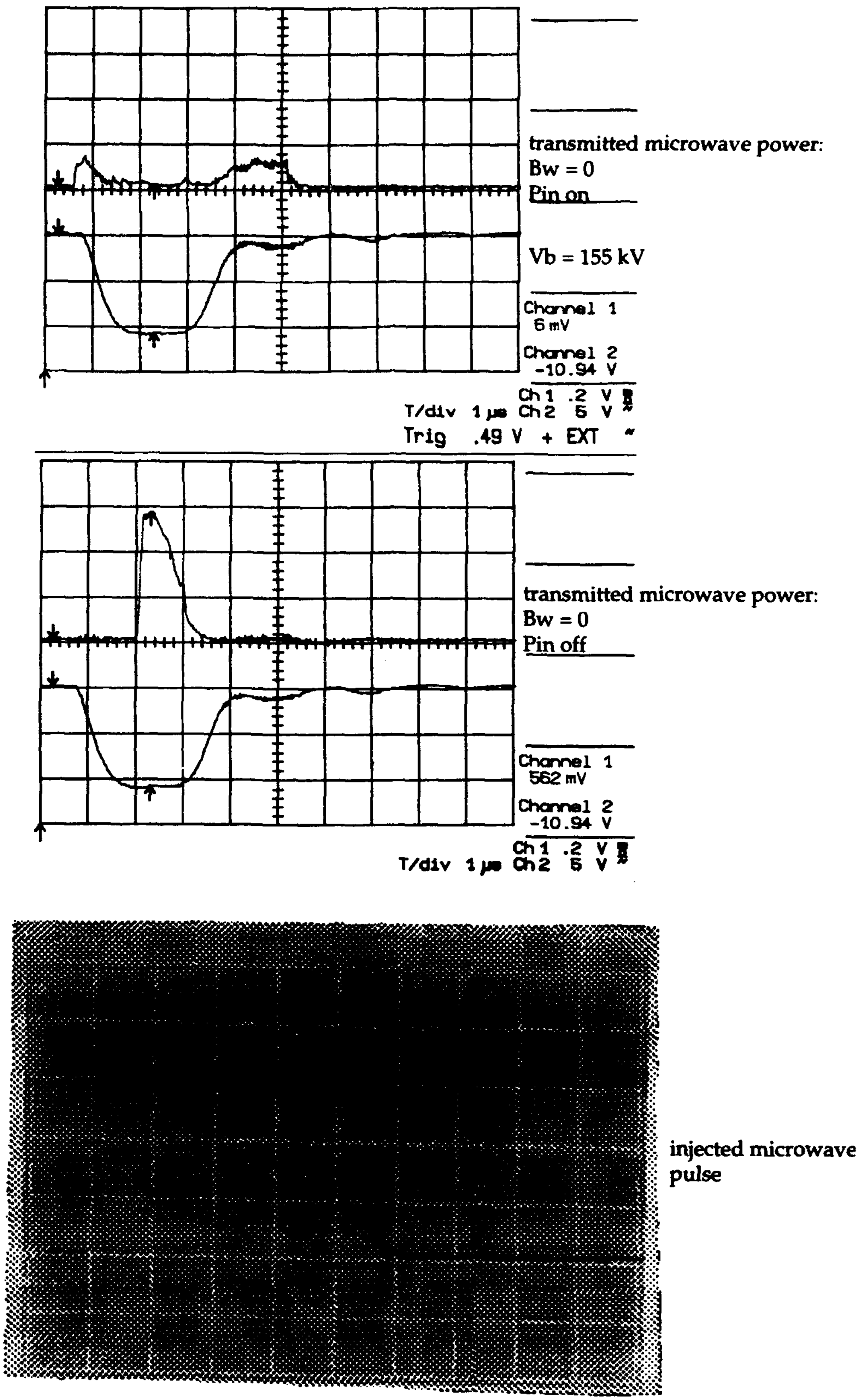


Figure 3.10. Suppression of oscillation with applied power.

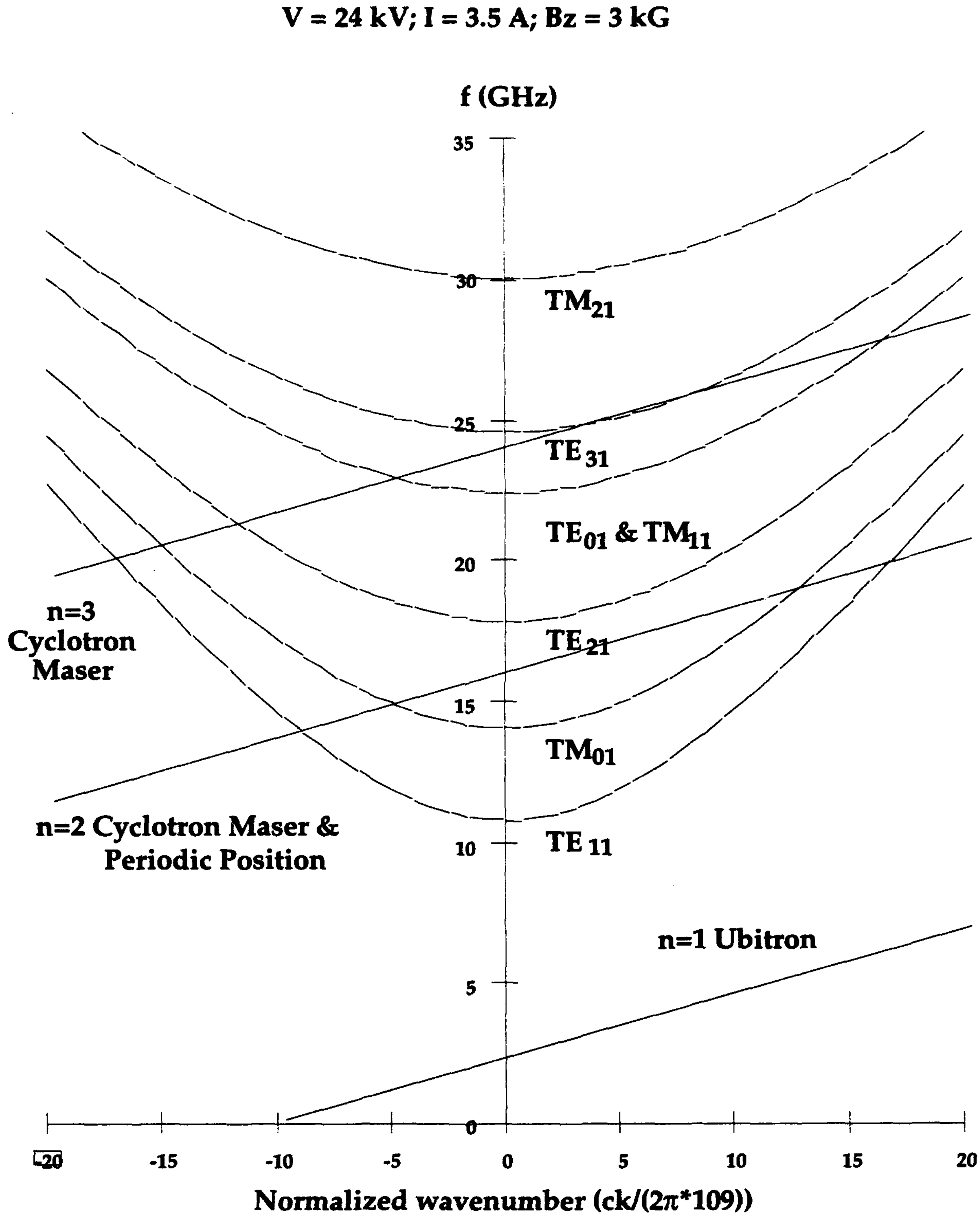


Figure 3.11. Uncoupled dispersion curves: V=24 kV, I=3.5 A, Bz=3 kG, Bw=0 G.

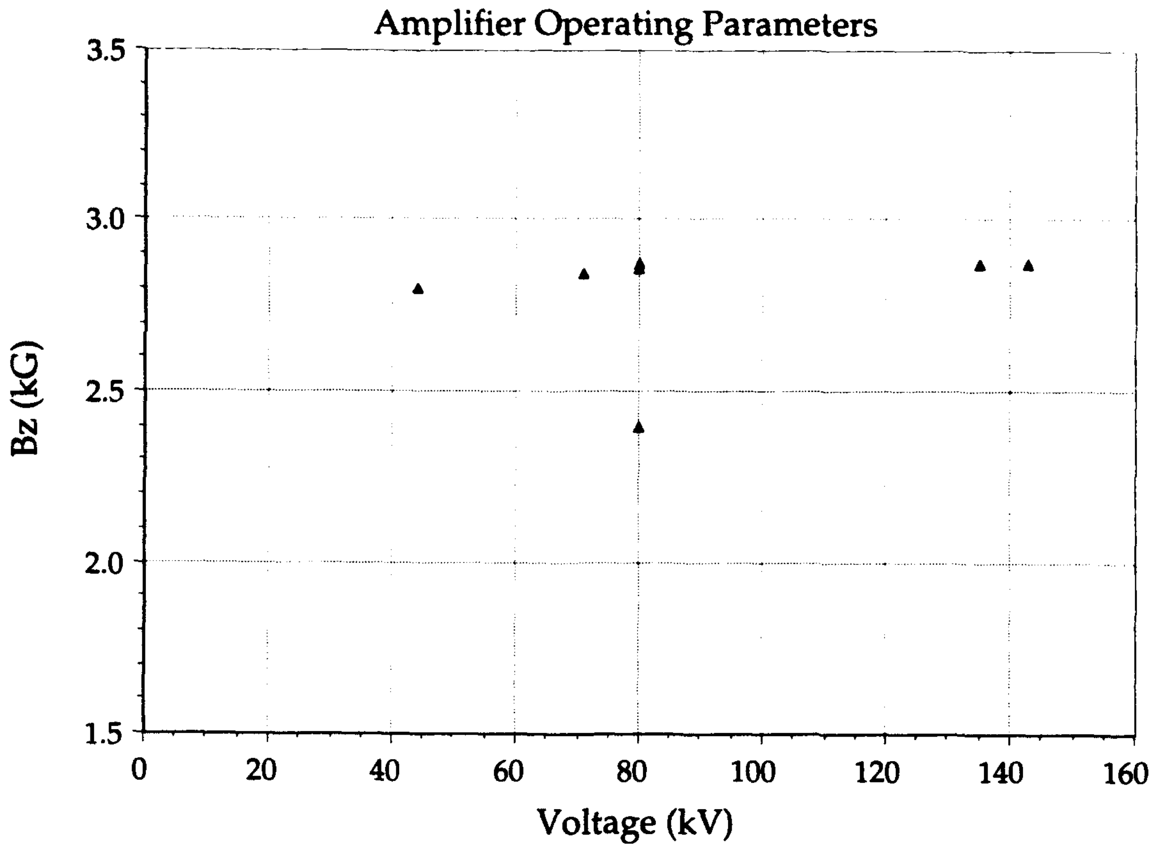
Finally, parameters for measured rippled beam amplifier and oscillator operation are summarized in Fig. 3.12. Note that this figure does not indicate that either mode of operation cannot occur at parameter values other than shown, only that measurements were made at these parameter sets.

3.1.4 Post Wiggler Meltdown Results.

In addition to limited time, further investigation of rippled beam amplifier and oscillator performance was curtailed by wiggler failure caused by coolant loss. Attempts to repeat previous results after wiggler repair were not particularly successful. No significant amplification was observed. Low amplification was measured, depending on the trim coil position relative to the cathode, as shown in Figs. 3.13-14. The upper trace in each set of Fig. 3.13 is the beam voltage; the lower trace of the upper set is the transmitted microwave power without wiggler field, and the lowest trace is the transmitted signal with the wiggler field. The upper trace of the upper set in Fig. 3.14 is the modulator current; the next trace is the output microwave signal from the HPA, followed by the injected and transmitted beam currents. Only ~ 1.5 dB of gain is measured, if the output frequency is the same as input; no frequency measurements were made. Similar measurements for different parameters are shown in Fig. 3.15, where the second and fourth traces are the transmitted signal without and with the wiggler field, respectively, and the sixth trace is the output pulse from the HPA. Gain could be increased to ~ 3.5 dB by reversing the trim coil polarity at higher beam voltage.

Oscillation characteristics were also slightly altered following wiggler failure, typically requiring a slightly higher axial magnetic field. At sufficiently high B_Z, no wiggler field was required for oscillation. Fig. 3.16 shows such a case for the following parameters, V = 35 kV, I = 5 A, B_Z = 3.2 kG, and B_W = 0 G. As observed for pre-meltdown cases, adding a sufficiently strong wiggler field could, in fact, suppress oscillation. As shown by traces two and four of Fig. 3.17, where a 145 G wiggler field reduced oscillation power.

The frequency of oscillation, when measured, was typically 17.9 GHz, as in the case: V = 35 - 40 kV, I = 4 - 6 A, B_Z = 3.2 kG, and B_W = 0 G. At higher beam



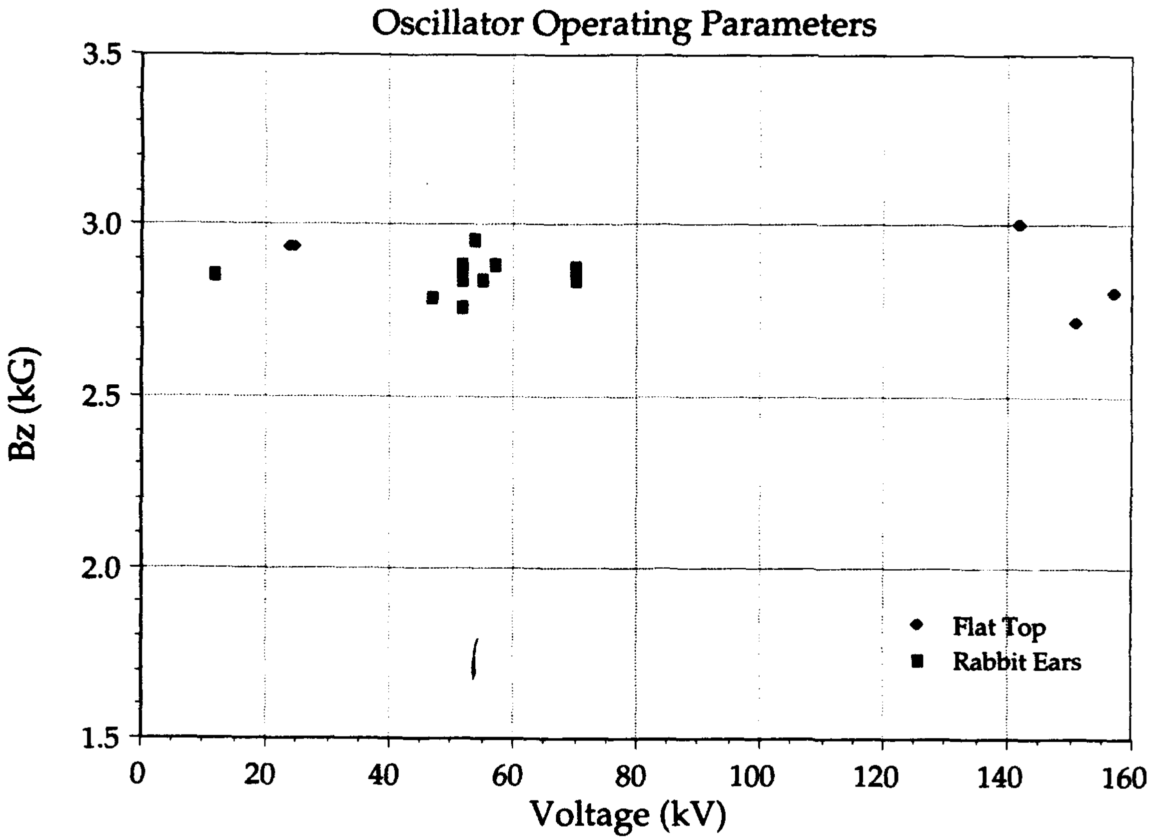


Figure 3.12. Parameters of measured rippled beam amplifier and oscillator operation.

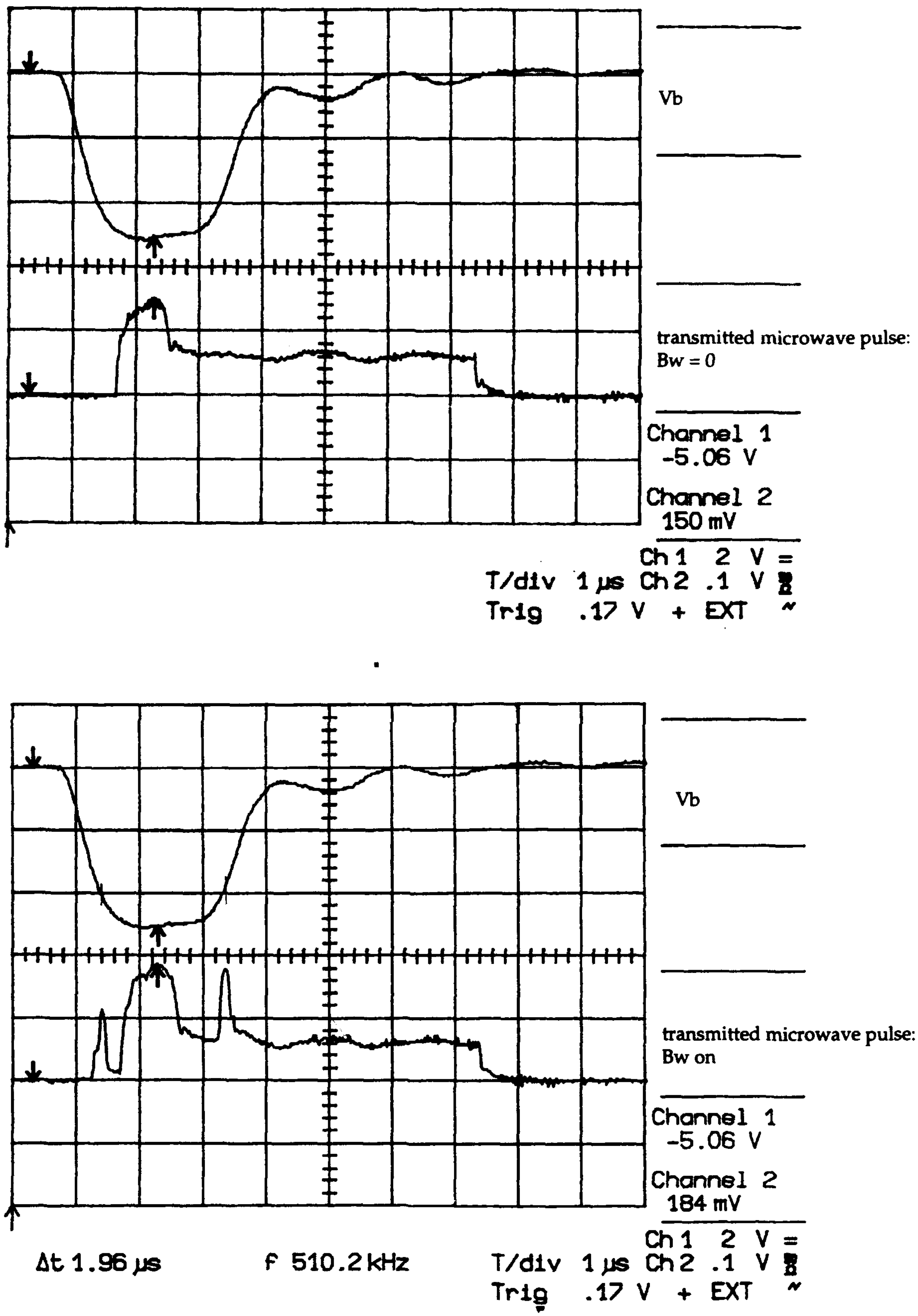


Figure 3.13. Post wiggler meltdown amplifications: beam voltage and transmitted microwave power.

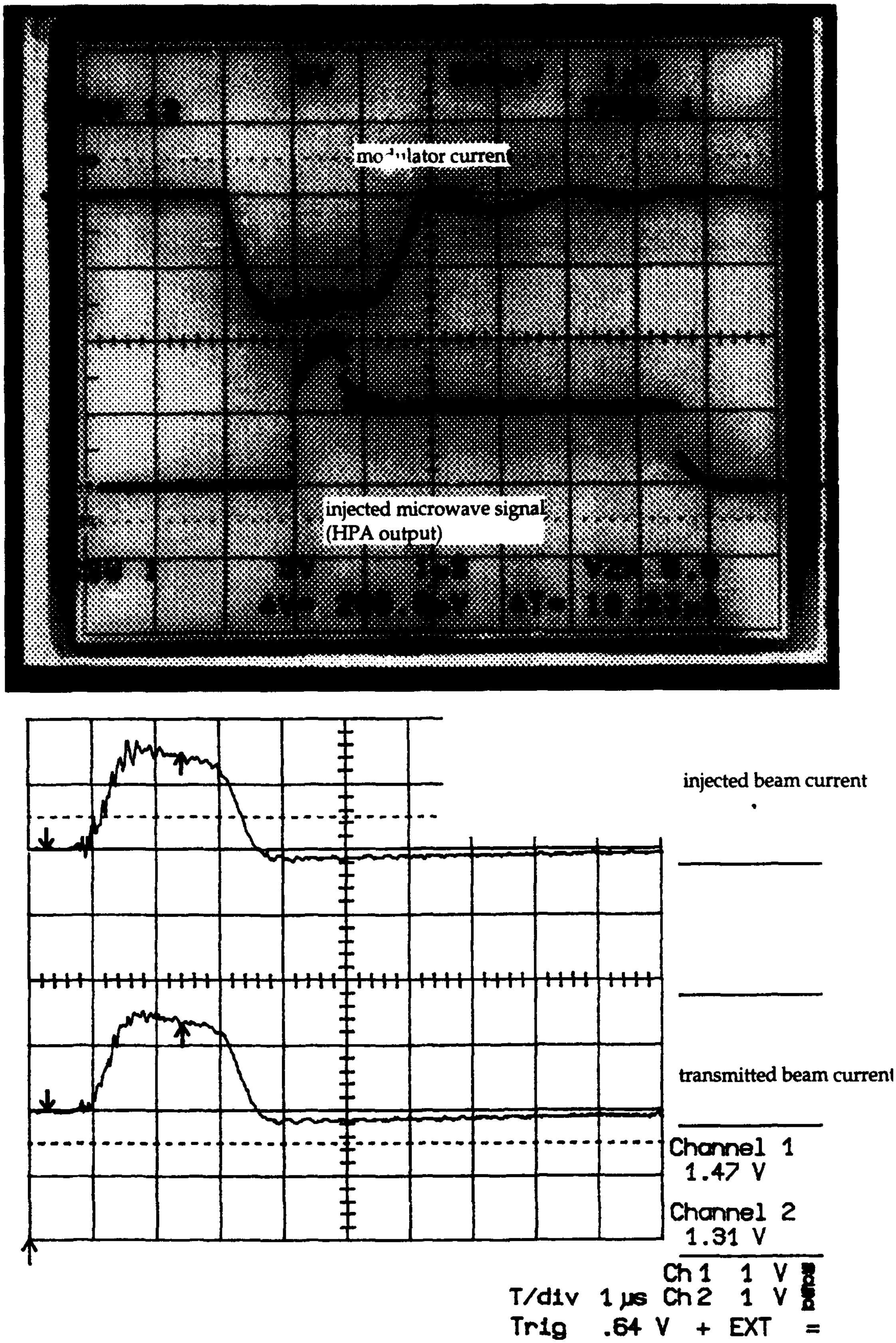


Figure 3.14. Post wiggler meltdown amplification: modulator current, HPA output, beam current.

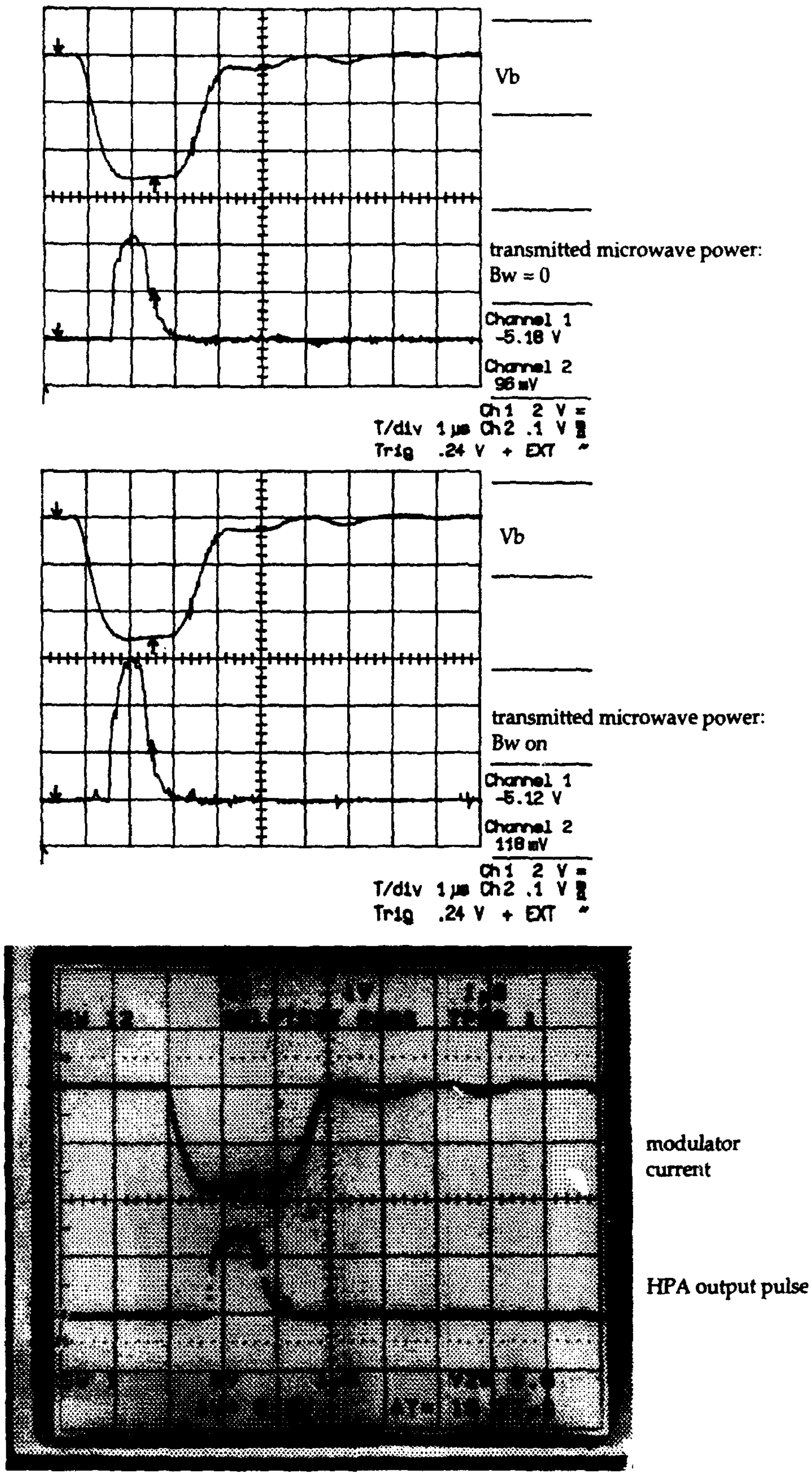
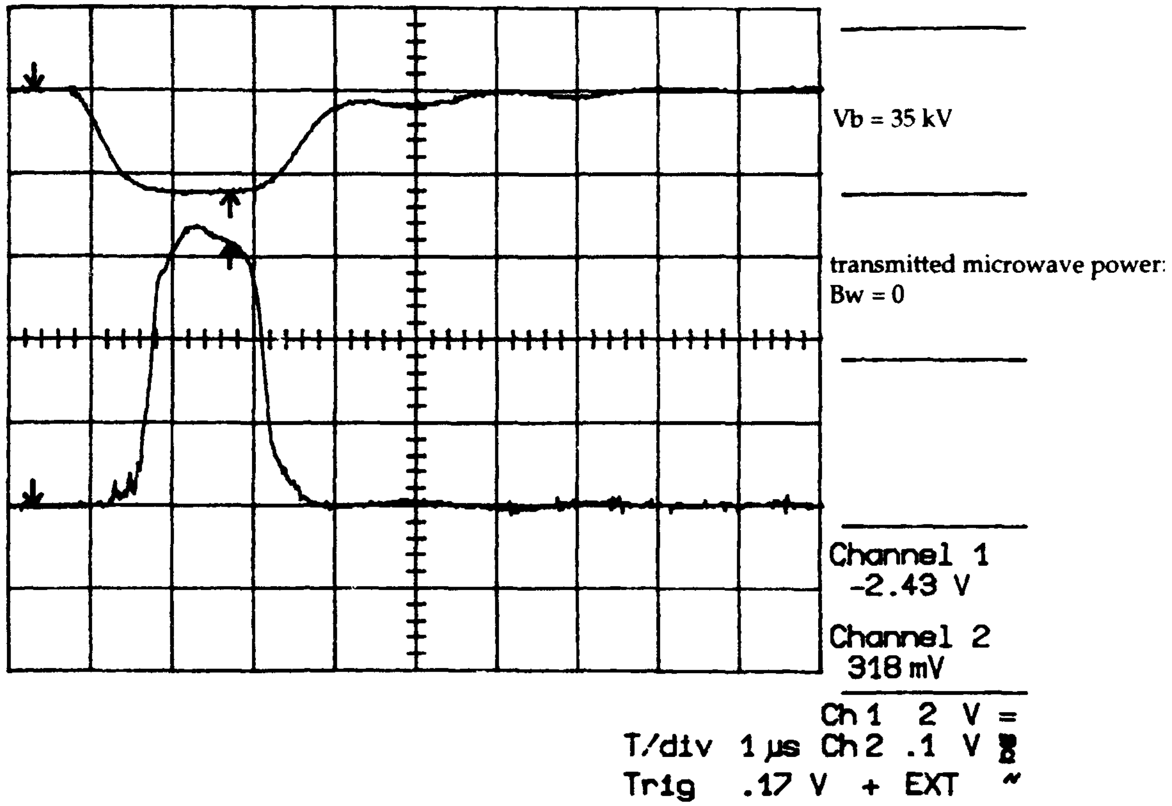


Figure 3.15. Post wiggler meltdown amplification: different parameter set.



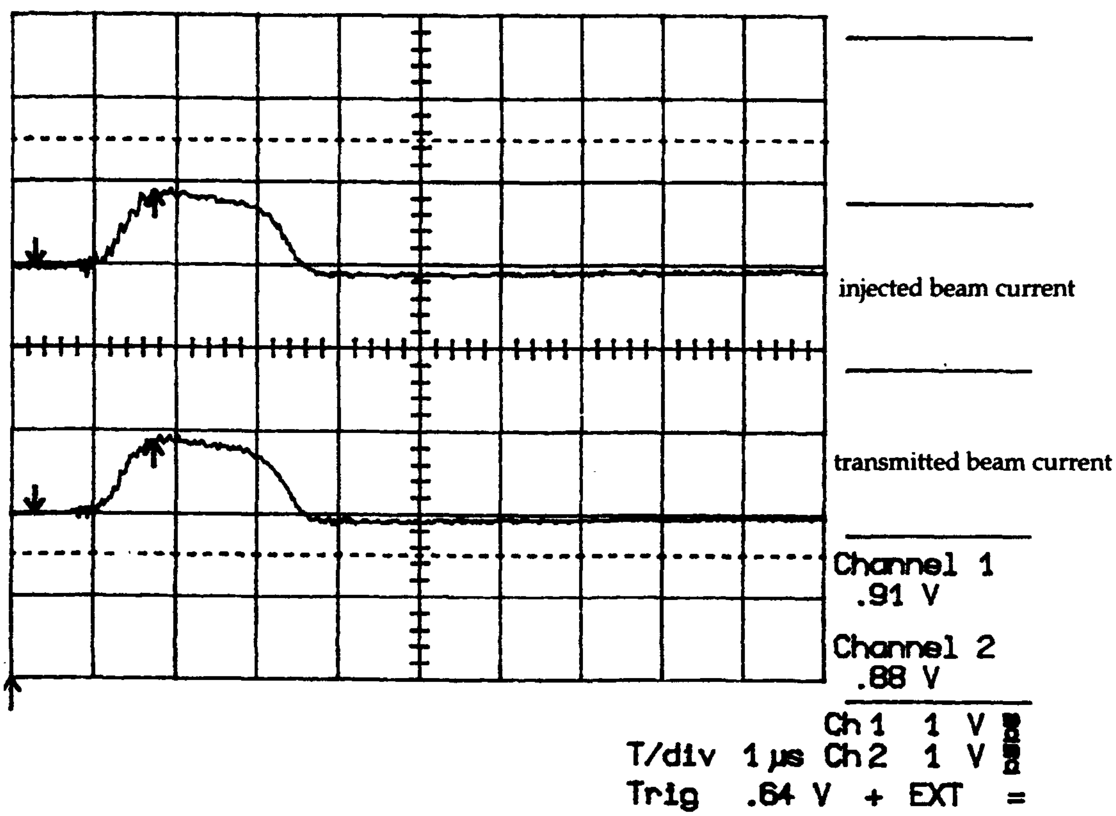


Figure 3.16. Post wiggler meltdown oscillation: V=35 kV, I=5 A, Bz=3.2 kG, Bw=0 G.

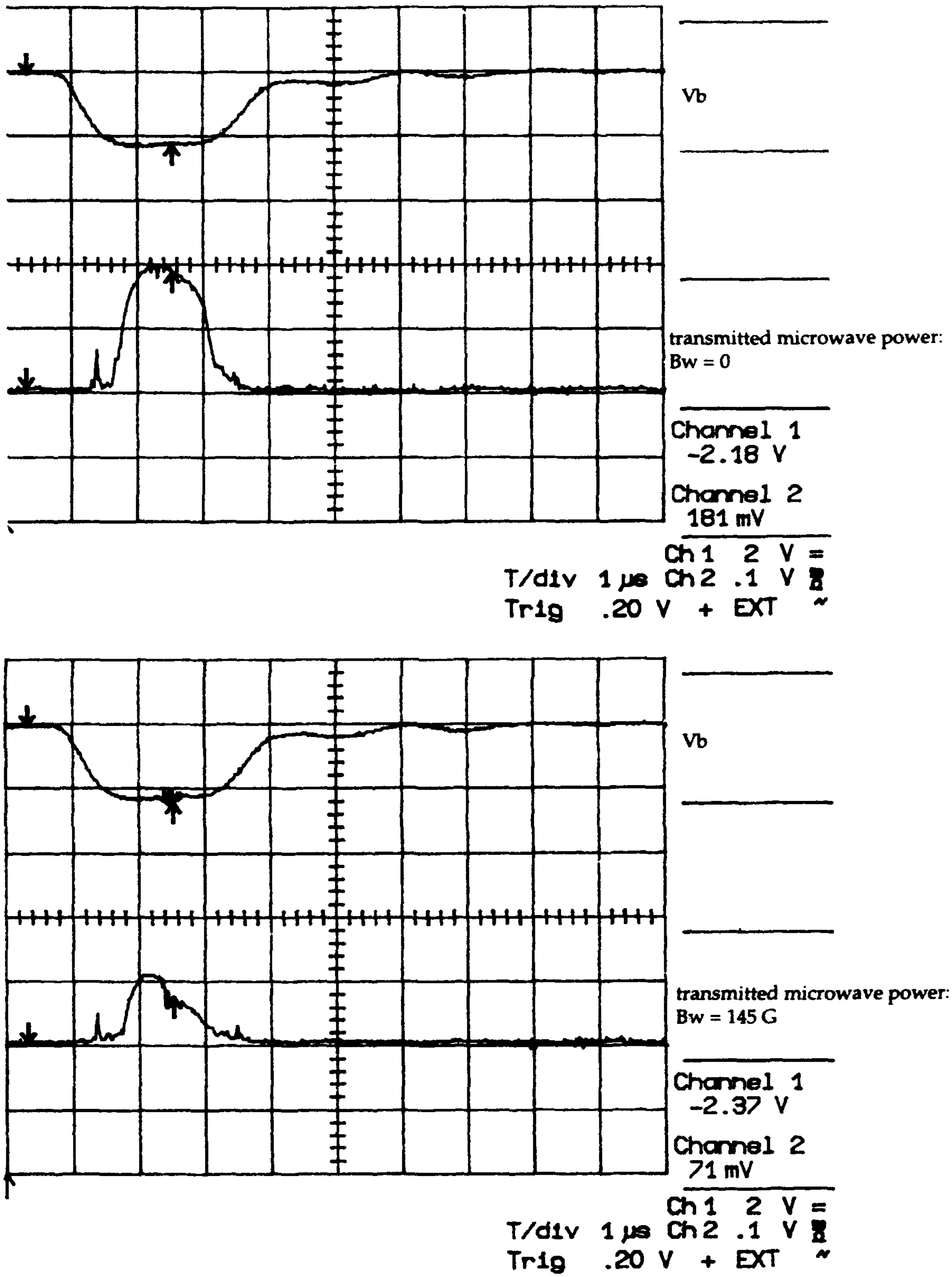


Figure 3.17. Post wiggler meltdown oscillation: suppression of oscillation with wiggler field.

voltages, this oscillation frequency was also measured at less than the maximum axial field with the addition of the wiggler field; V = 144 - 195 kV, I = up to 30 A, B_Z = 2.9 - 3.2 kG, and B_W = 89 - 140 G. Depending on mode and polarization, the output power could be as high as 60 kW under these conditions.

While both amplification and oscillation were observed following wiggler failure, the most plausible reason for the inability to reproduce earlier results is a much altered wiggler field resulting from in-situ repairs. To repair wiggler damage without breaking vacuum, the wiggler was rewound in place with four 12-ga. round wires replacing ten edge-wound rectangular wires. To compensate for the reduced gauss/amp, higher current repetive pulse operation was required. Although the wiggler field was not measured until after rippled beam characterization was concluded, the aluminum winding form seriously altered the wiggler field entrance profile under pulsed operation (see Sec. 2.4.).

3.1.5 Rippled Beam Summary.

Both the amplifier and oscillator results discussed above are significant in that respectable gain, bandwidth, and power were obtained with (probably) second harmonic operation. T o repeat, a peak gain of 24 dB, bandwidth > 7%, and output power ~ 25 kW were obtained at beam voltages 10 - 30% of the design value. Recall that for the interaction mechanisms of interest, fundamental mode operation is below waveguide cutoff. Recall, also, that the experimental configuration was not designed with rippled beam operation under consideration. Further experimental and theoretical investigation of this mode of operation could lead to significant performance improvements.

3.2 LAMINAR BEAM PERFORMANCE.

3.2.1 Introduction.

Rippled beam operation, discussed in the previous section, utilized, in part, a cathode surface located in the relatively rapidly decreasing axial magnetic field outside of the solenoid with a four-inch diameter opening in the pole piece. This enabled a relatively modest gun trim coil to cancel the axial field at the cathode surface, thus generating a highly rippled electron beam resulting from electrons with essentially zero canonical angular momentum being injected into a uniform axial magnetic field. However, for conventional ubitron operation, this type of electron beam is not appropriate; a laminar electron beam is required.

To achieve a laminar electron beam, the pole piece aperture was enlarged to approximately seven inches in diameter, and the gun repositioned. With these modifications, SCRIBE electron trajectory calculations indicate that the axial velocity spread of the beam as it exits the input coupler is reduced to the range of 0.26% to 0.023%, depending on trim coil current. Although the extracted beam quality was not directly measured, a comparison of code calculations of total diode current and transmitted beam current shows good agreement with measured values.

In common with the earlier rippled beam measurements, experiments with a laminar beam were required to use the pulsed wiggler, with the then unknown field profile. This is discussed later in more detail when comparing experimental results and theoretical predictions.

To characterize ubitron amplifier performance, the device was operated over as large a parameter space as possible, within the limitations of beam voltage, field amplitudes, and microwave frequency range and power. The parameter range over which the present experiment has been operated are presented below, in comparison with design values.

	Present	Design
Voltage (kV)	190-250	250
Current (A)	0-37	30/100
Beam Radius (cm)	0.4	0.4
Pulse Length (μs)	1	1
Repetition Rate (Hz)	3-30	1-100
Wiggler:		
Period (cm)	2.54	2.54
Entrance (periods)	5	5
Uniform (periods)	12	12
Exit (periods)	3	3
Pulsed Field (G)	575	1500
DC Field (G)	140	500
Solenoid (kG)	1.8-2.8	1-3.2
Frequency (GHz)	13.5-17.4	12.4-18

In order to achieve high gain, efficiency and bandwidth simultaneously, operating parameters have been chosen to produce a grazing intersection of the wiggler-shifted negative energy space-charge wave with the TE_{11} circular waveguide mode. Uncoupled dispersion curves for two representative parameter sets are shown in Figs. 3.18-19, including lines for the fundamental ubitron interaction and the first two gyrotron harmonics. Note that a broad intersection is achieved for the ubitron line and that possible gyrotron interactions are well separated and would be identifiable by both frequency and mode. Grazing intersection results in slightly different characteristics than those usually associated with FEL's. Voltage tuning is negligible while instantaneous bandwidth becomes very large. Lowering the voltage or increasing the wiggler field beyond a certain point results in narrowing and then decoupling of the gain profile, while raising the voltage eventually results in a double-peak profile with decoupling in the center of the band.

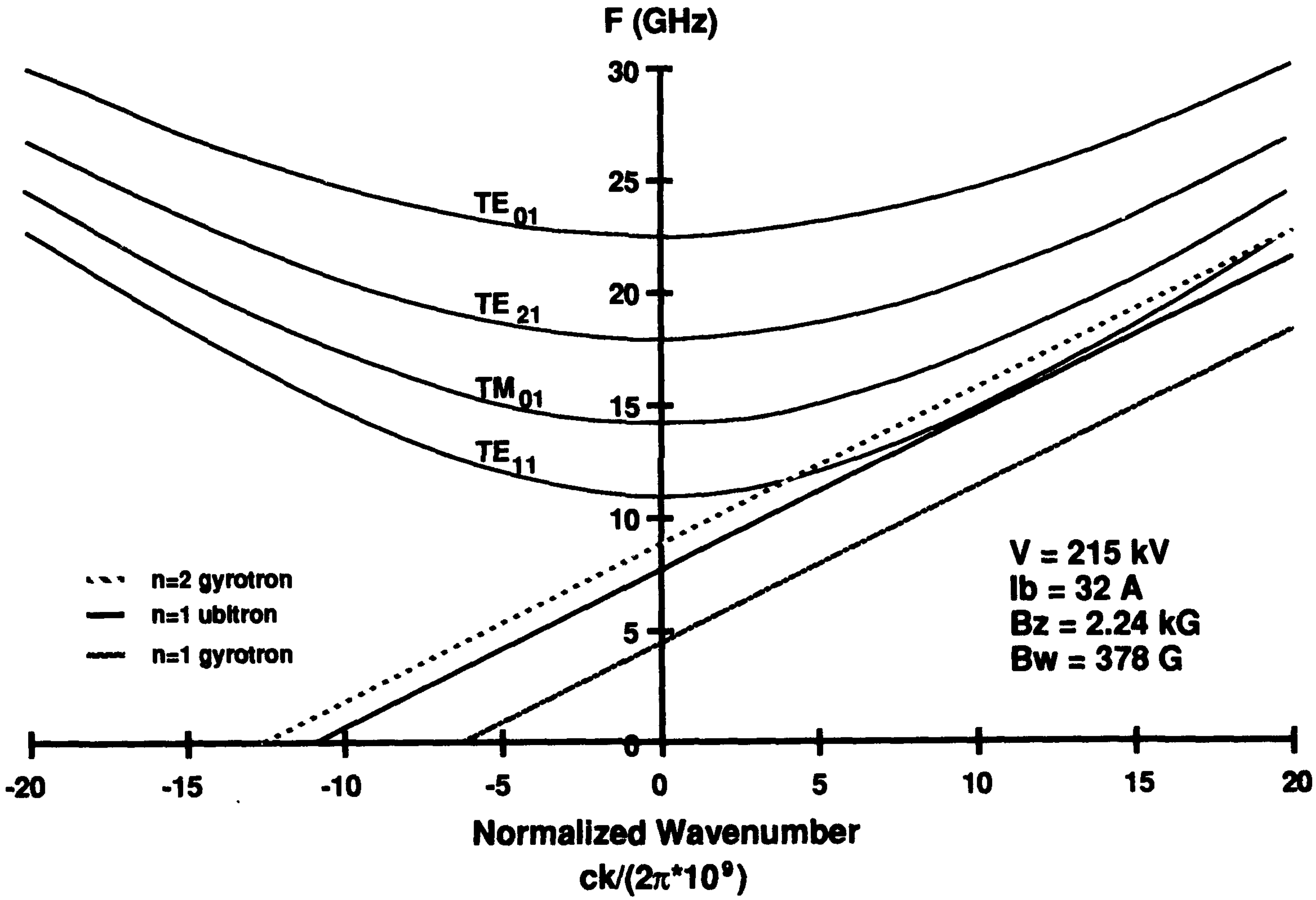


Figure 3.18. Uncoupled dispersion curves: V=215kV, I=32A, Bz=2.24kG, Bw=378G.

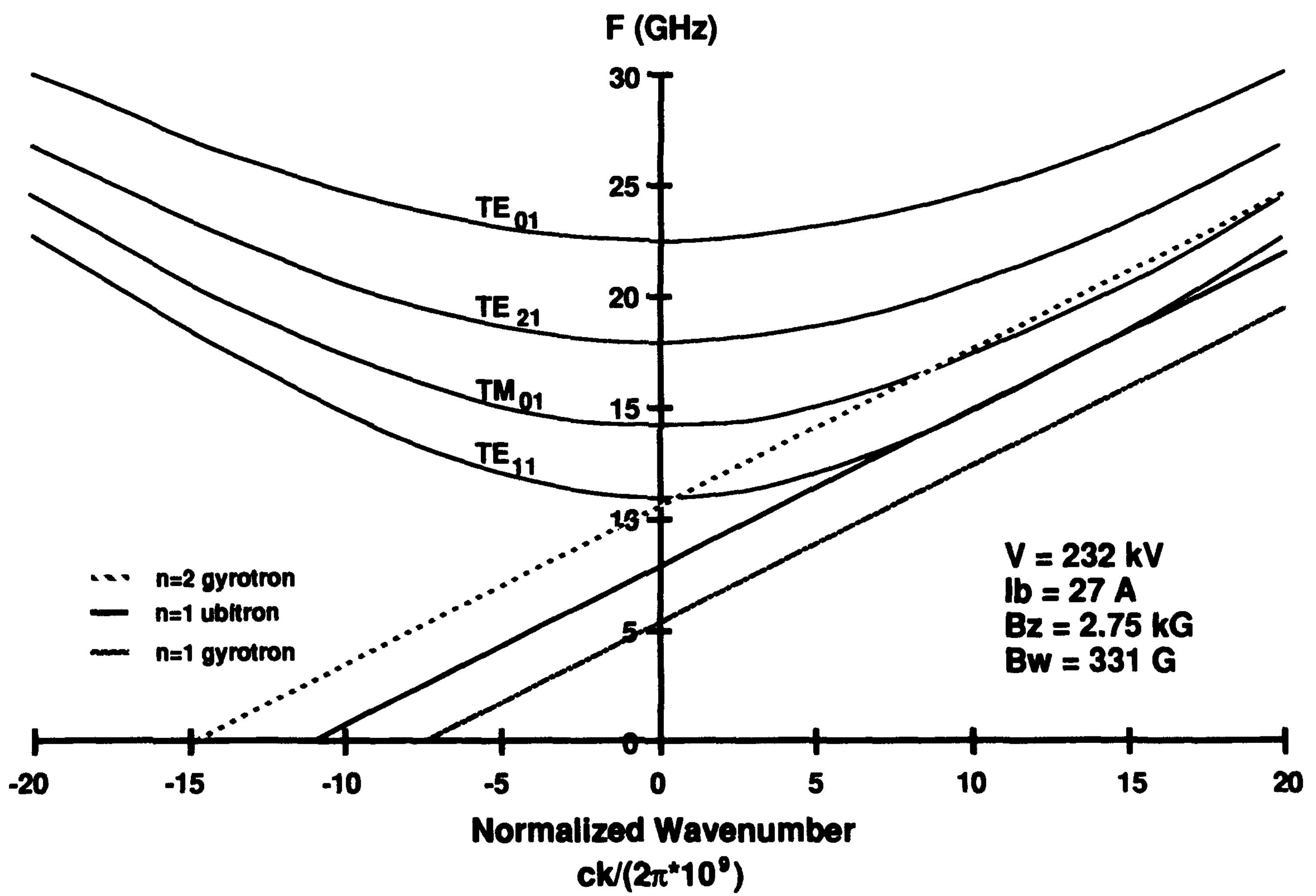


Figure 3.19. Uncoupled dispersion curves: V=232kV, I=32A, Bz=2.75kG, Bw=331G.

However, operation near grazing intersection should have a beneficial effect on issu^- such as phase sensitivity to voltage and wiggler variations.

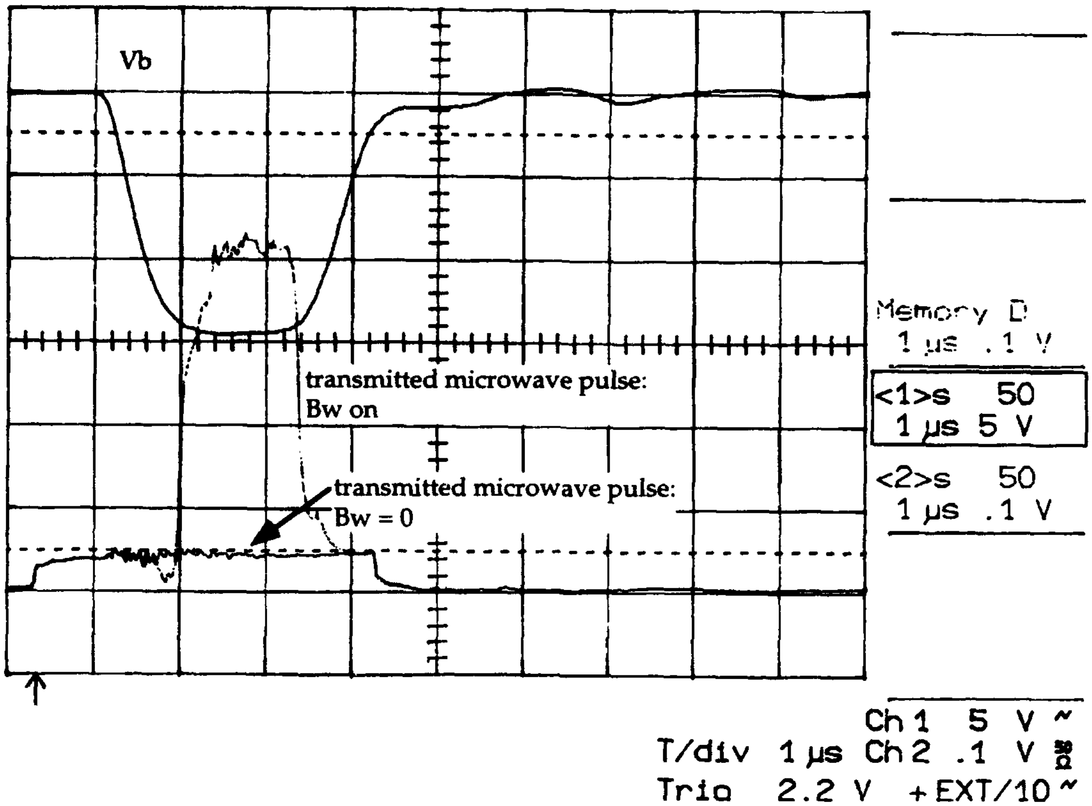
3.2.2 Amplifier Performance.

Following the laminar beam solenoid modifications, a series of amplifier measurements were made. Microwave performance was measured as a function of input signal polarization, microwave drive power and frequency, wiggler field, and injected current. One of the unique features of this experiment is a flexible input coupling scheme providing the capability to launch circularly and linearly polarized waves, as well as selected waveguide modes. The microwave results presented below were measured using a left circularly polarized input wave. As predicted by theory, very little or no gain was observed using right circular polarization. Theory also indicates that the combination of a helical wiggler field and a circularly polarized wave will yield the highest gain for a given input power and wiggle velocity.

3.2.2.1 Typical Waveforms.

Typical modulator voltage and microwave output coupler traces are shown in the upper trace and the two lower traces, respectively, in the upper graph of Fig. 3.20. Two microwave traces are shown, the transmitted microwave-driver signal without the wiggler (lower) and an amplified signal with wiggler turned on. The width of the microwave driver pulse and the overlap with the voltage pulse can be adjusted as desired. Although the portion of the microwave driver pulse before the voltage flat top could be used to determine gain, this does not account for several factors such as beam loading of the input coupler. Hence, gain is determined from the output power measured with the wiggler turned off and the output power measured with the wiggler turned on, all other factors being held constant. Both a digital oscilloscope and a peak power meter are used for these measurements.

Pulsed operation of the wiggler is shown in the upper trace of the lower graph of Fig. 3.20, and the transmitted beam current, with and without the wiggler on, are shown in the lower two traces. Note that the wiggler current is flat during the beam current pulse. The beam current traces show that the



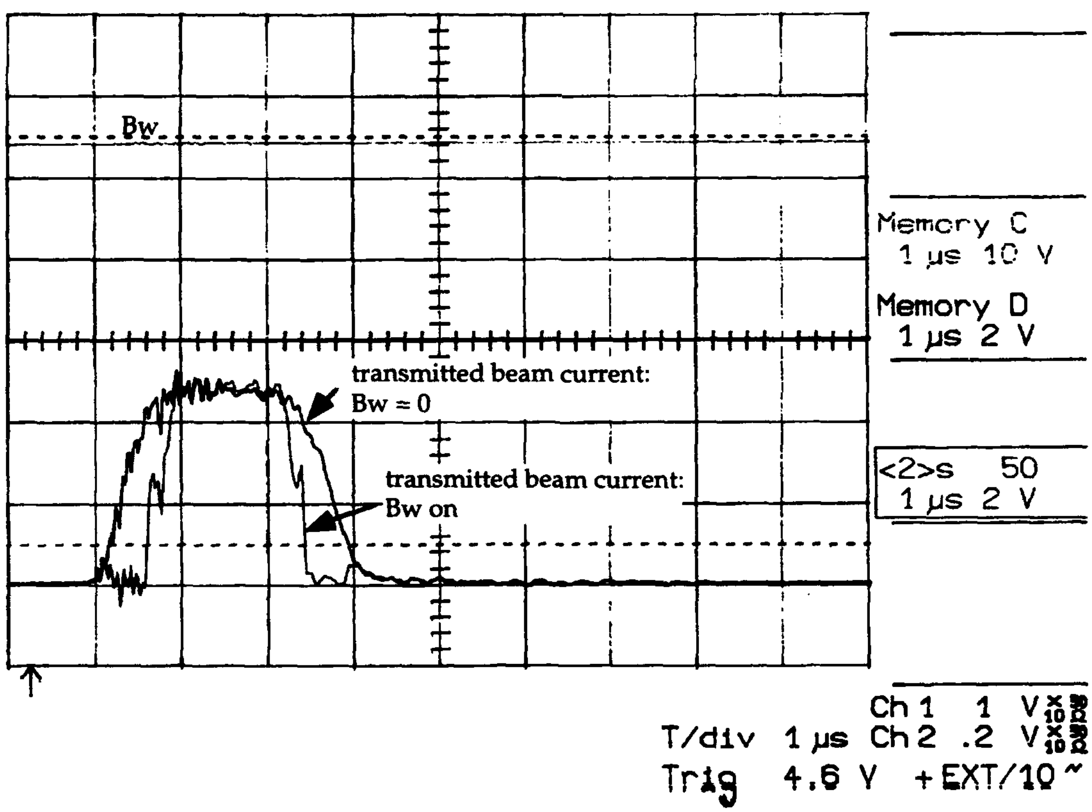


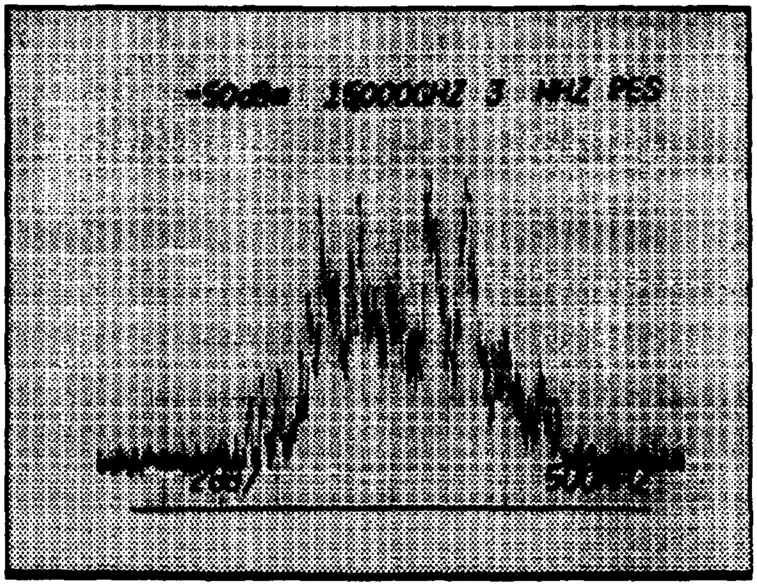
Figure 3.20. Typical amplifier waveforms.

electrons are initially on stable, Group II orbits at the beginning of the voltage pulse. As the voltage rises, the electrons move toward and through gyroresonance, and then onto stable, Group I orbits. A dip in transmitted current occurs on the rise and fall of the voltage pulse due to instability at gyroresonance. Note that the current is flat during the voltage flat top and that the wiggler has little effect on the transmitted current level. Other factors monitored on each pulse are diode current, injected current, transmitted current, wiggler current and calorimeter temperature. It should also be noted here that amplifier operation under laminar beam conditions is more stable than under rippled beam conditions.

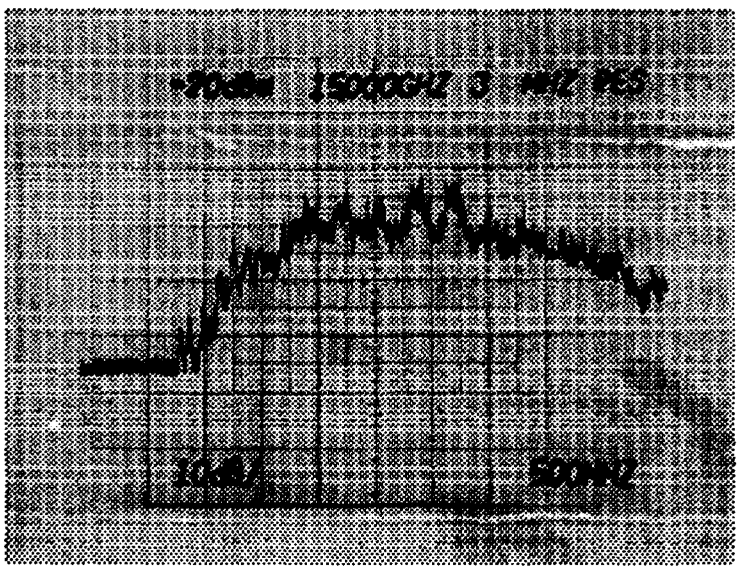
3.2.2.2 Gain Measurements.

All gain measurements are presently limited in bandwidth by the output of the high-power driver which drops off at 13.5 GHz and by the performance of the input coupler which falls off around 17 GHz. The measured values typically show a double-peak profile over the 13.5-17.4 GHz band (≥ 25% bandwidth), although a fairly flat gain profile has been measured with parameters selected for barely grazing intersection. The peak measured gain is approximately 19 dB at 17 GHz. For a different parameter set, a peak gain of 17 dB was measured at 13.5 GHz. Assuming that gain occurs mainly over the 12 uniform periods in the wiggler, this translates into approximately 1.25 dB per free space wavelength. This value, achieved at less than optimum operational parameters, is an improvement over the 0.5-0.7 dB per free space wavelength achieved in Phillips' and other more recent experiments [1-3].

There was some concern that noise from the intermediate amplifier (-30 dBc) would interfere with amplifier measurements. The spectrum of the amplified noise transmitted through the ubitron (without beam) is shown in Fig. 3.21, including the frequency responses of the high power amplifier and the ubitron input and output couplers. For the upper trace, the sweeper was connected, but switched to Standby. The input to the intermediate amplifier was shorted for the lower trace. High-power, adjustable frequency filters that would have eliminated this problem were not available for insertion between the intermediate and high power amplifiers. To insure that amplified noise from the intermediate amplifier was not influencing amplifier results, a Ferretec tracking



Amplified noise transmitted through ubitron (no beam): intermediate amplifier input 'open'



Amplified noise transmitted through ubitron (no beam): intermediate amplifier input shorted

Figure 3.21. Amplifier noise transmitted through ubitron.

filter was inserted before the output detector for several parameter sets. No differences were noted in gain profiles, with or without the filter, as shown in Fig. 3.22 for a typical case, time-resolved.

Measured gain as a function of frequency is shown in Figs. 3.23-26 for seven combinations of beam voltage, current, axial and wiggler magnetic fields. The solid line through the data points is smooth fit to assist visualization of the gain profile. A variety of profiles is possible: flat, peaked at midband frequencies, or peaked at band extremes. Several additional typical characteristics are noted. Limited tuning of the axial magnetic field has shown no strong dependence of the interaction on field value. When the experiment is adjusted for peak gain, lowering the voltage results in lower gain and a reduced bandwidth. The experiment has been operated as low as 190 kV where 3-4 dB total gain was observed in the center of the band. Raising the voltage also results in lower gain at center band as the ubitron and waveguide dispersion uncouple. In this case the amplified microwave signal hollows out as the gains at the edges of the voltage pulse are higher than the gain at the voltage flat top. At present operating voltages, the wiggler field cannot be increased sufficiently (without dispersion uncoupling) to produce the 25-30 dB gains that would be required to saturate the experiment with available microwave input power. Hence, all measurements are in the small signal range, and the power out is directly proportional to power in.

3.2.2.3 Additional Measurements.

In addition to gain vs. frequency measurements, the dependences of gain on wiggler field (wiggler current) and beam current for fixed input frequency were also investigated. In the collective mode of operation, gain is proportional to beam α, the ratio of $v_\perp/v_\parallel$, which is directly proportional to the wiggler field. This relationship is approximate due to the wiggler-axial magnetic field gyroresonance and the increase in the wiggler field as the beam moves off-axis. Measurements of gain versus wiggler field are shown in Fig. 3.27 for several frequencies. The expected linear behavior is seen initially, but then the curves roll over and begin to drop as the ubitron line decouples from the waveguide mode. Note that the higher frequency curves drop off faster than the others

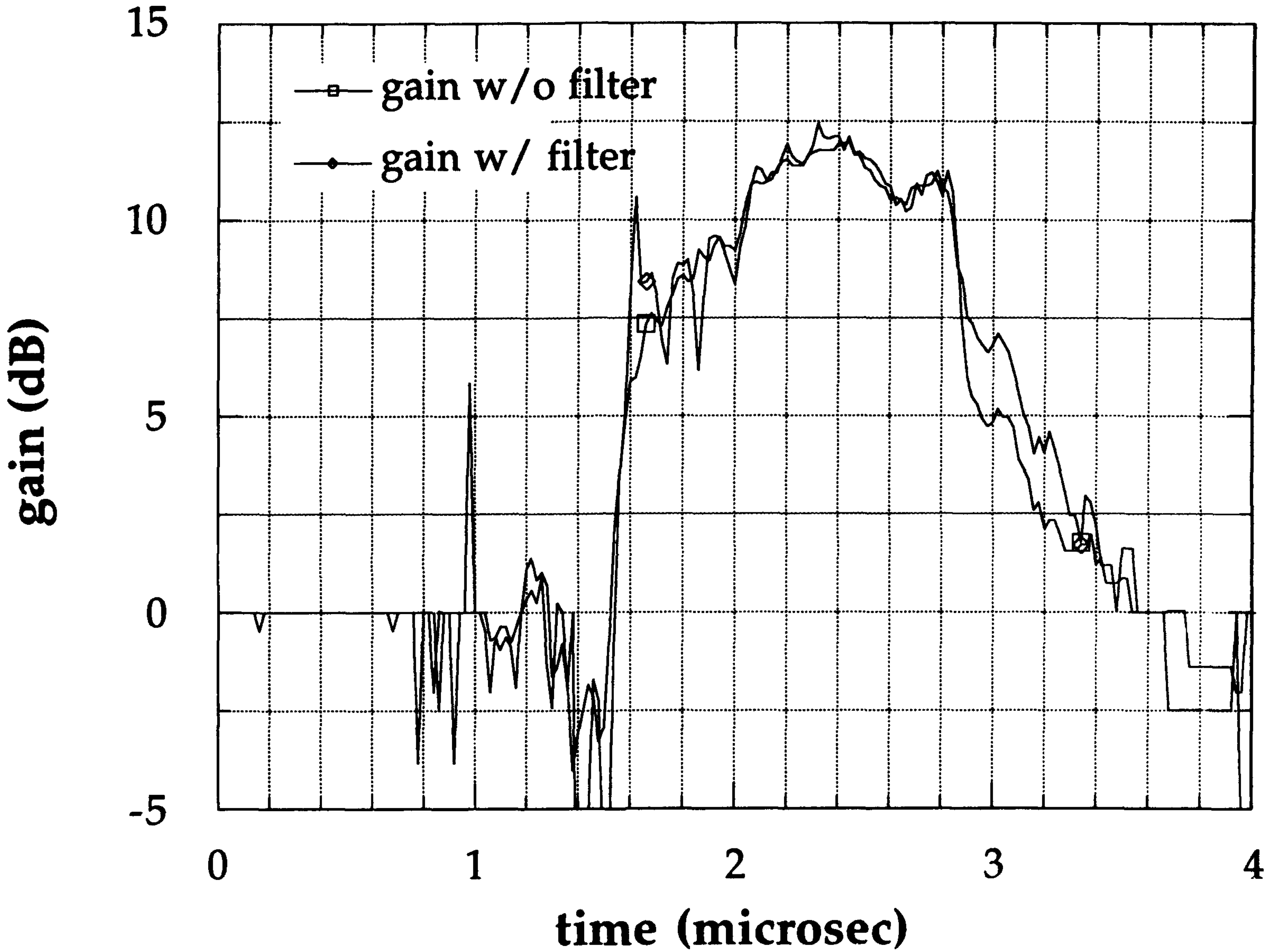
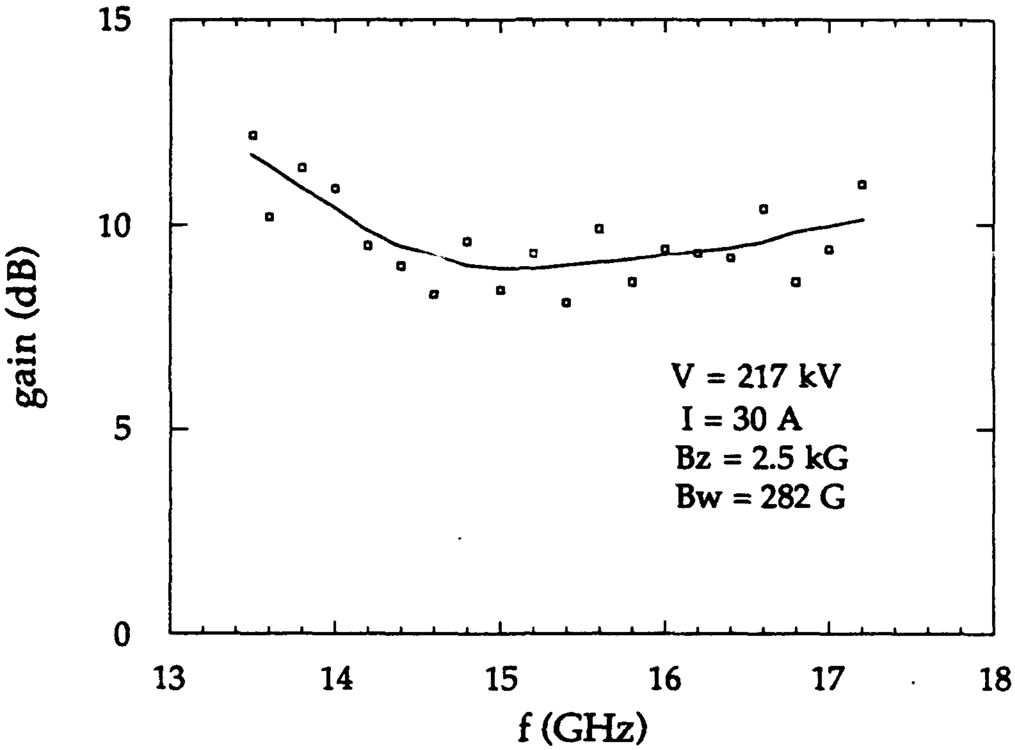


Figure 3.22. Gain temporal profiles; with and without filter.



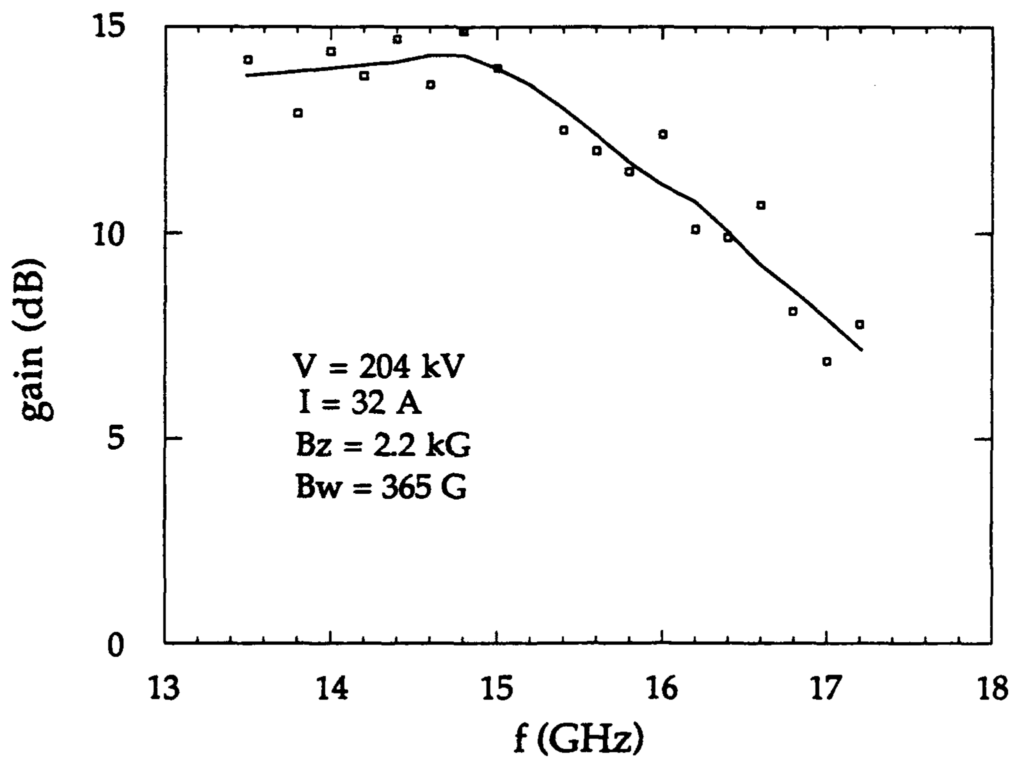
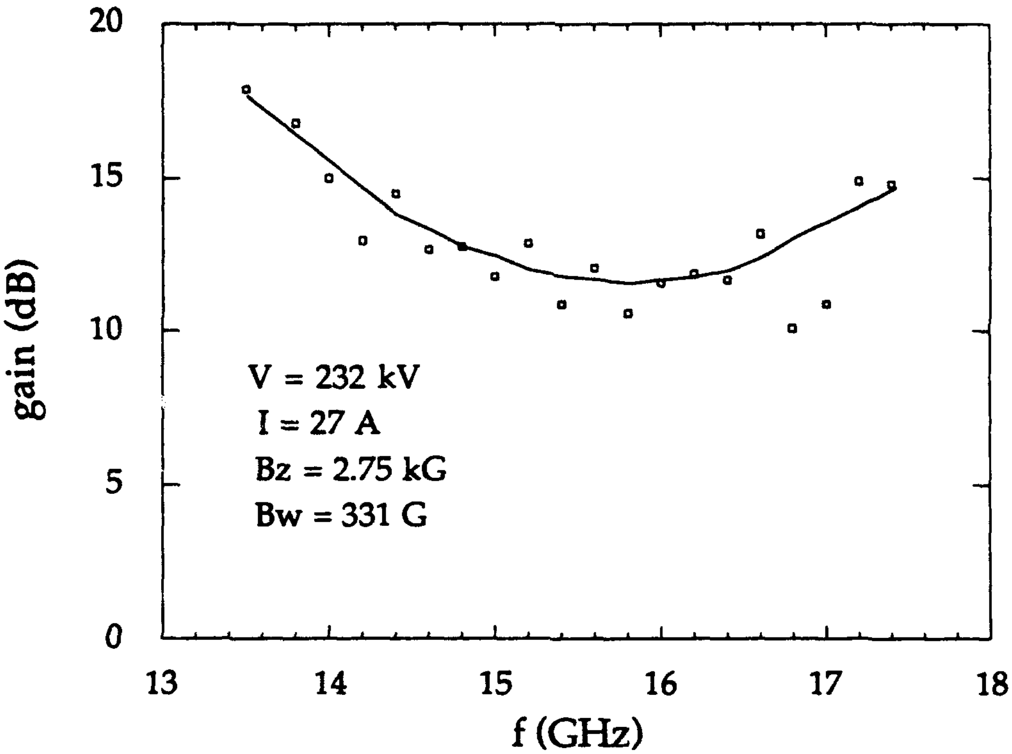


Figure 3.23. Small signal gain vs. frequency: parameter sets a and b.



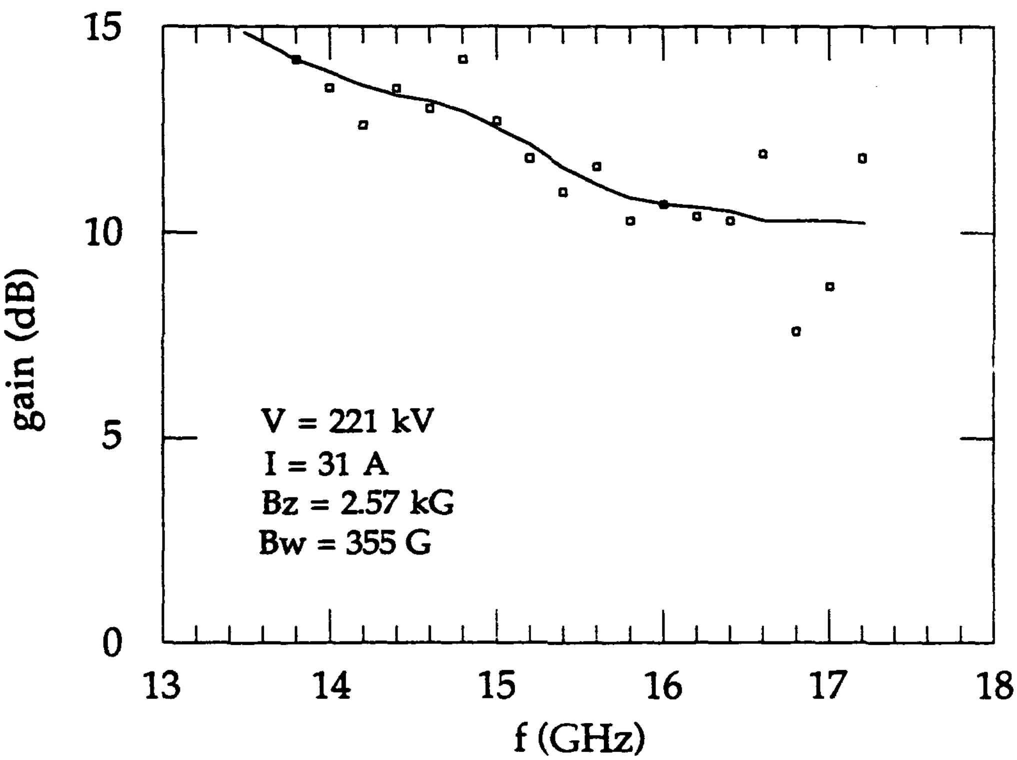
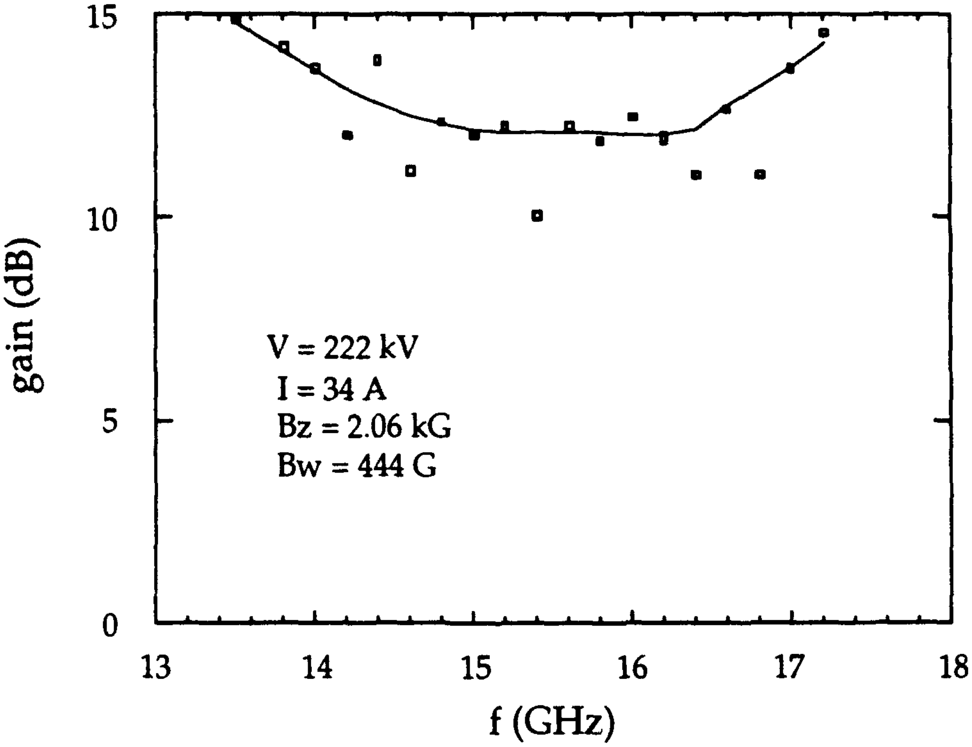


Figure 3.24. Small signal gain vs. frequency: parameter sets b and c.



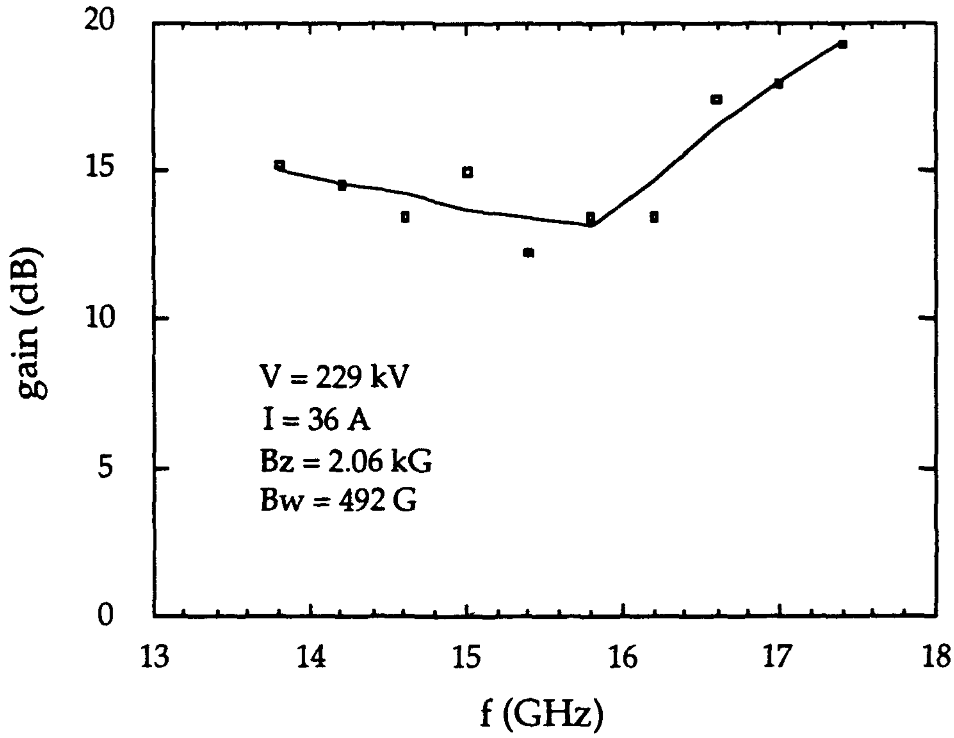


Figure 3.25. Small signal gain vs. frequency: paramter sets e and f.

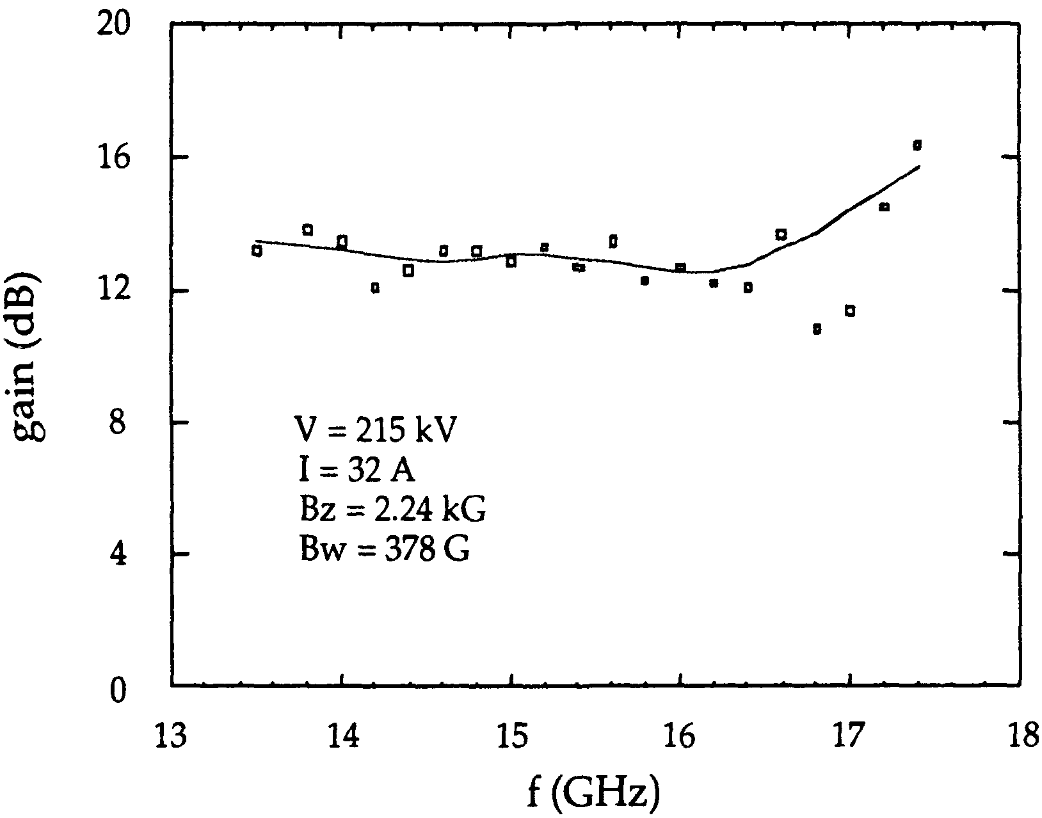
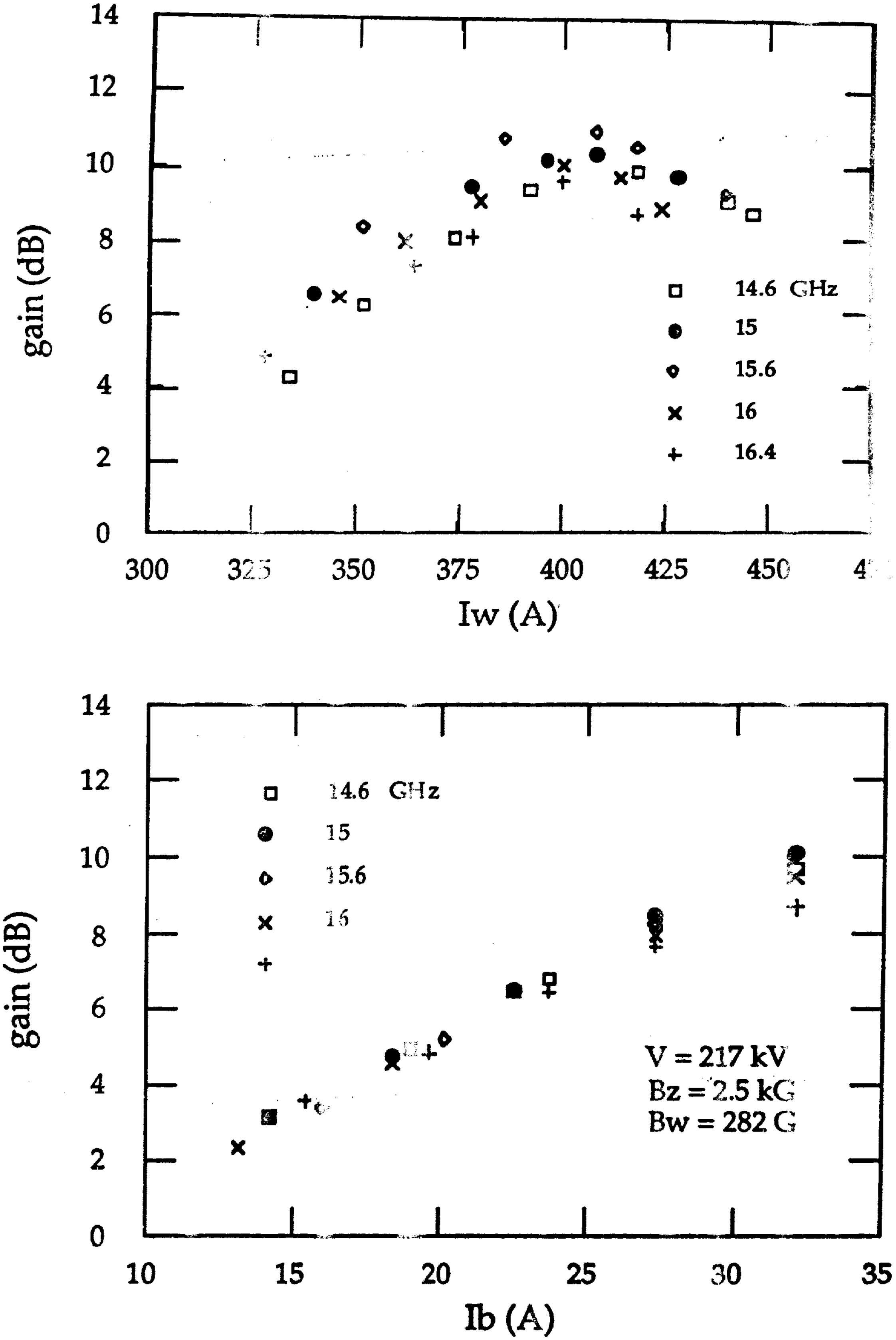


Figure 3.26. Small signal gain vs. frequency: parameter set g.



Fiugre 3.27. Small signal gain vs. wiggler field and beam current.

(which are nearer band center). This is expected since the gain profile narrows as the ubitron and waveguide dispersions uncouple.

Small signal gain as a function of injected beam current is also shown for several frequencies in Fig. 3.27. The theoretical dependence of the gain on injected current is fourth root in the collective mode. This is consistent with measurements, but the range of currents tested is not sufficient to discriminate between an $I^{1/4}$ (collective) or $I^{1/3}$ (strong-pump) dependence. Note again that the higher frequencies show a somewhat greater fall off. This is also due to gain profile narrowing and, in part, by the increase in the beam plasma frequency reducing the ubitron/waveguide dispersion overlap.

Output power dependence on input power was measured for only one parameter set: V = 213 kV, I = 30 A, B_z = 2.5 kG, B_w = 275 G, and f = 14.74 GHz. Saturation was not observed for this set, shown in Fig. 3.28, or for any other parameter set.

3.2.3 Analysis.

The experimental observations are compared with a fully three-dimensional nonlinear analysis and simulation of the ubitron/FEL [6p-9p] for this configuration. In this analysis, a set of coupled nonlinear differential equations is solved which describes the evolution of the trajectories of an ensemble of electrons as well as the electromagnetic fields. The nonlinear current which mediates the interaction is computed from the microscopic behavior of the electron ensemble by means of an average of the electron phases relative to the ponderomotive potential formed by the beating of the wiggler and microwave fields. No wiggler average is performed over the electron trajectories; rather, the orbits are integrated in three-dimensions using the Lorentz force equations. As a result, it is possible to model the injection of the electron beam into the interaction region with specified beam initial conditions. The beam energy spread within the interaction region is dependent on both the initial energy spread (arising from the gun and beam transport system) and the wiggler field gradient.

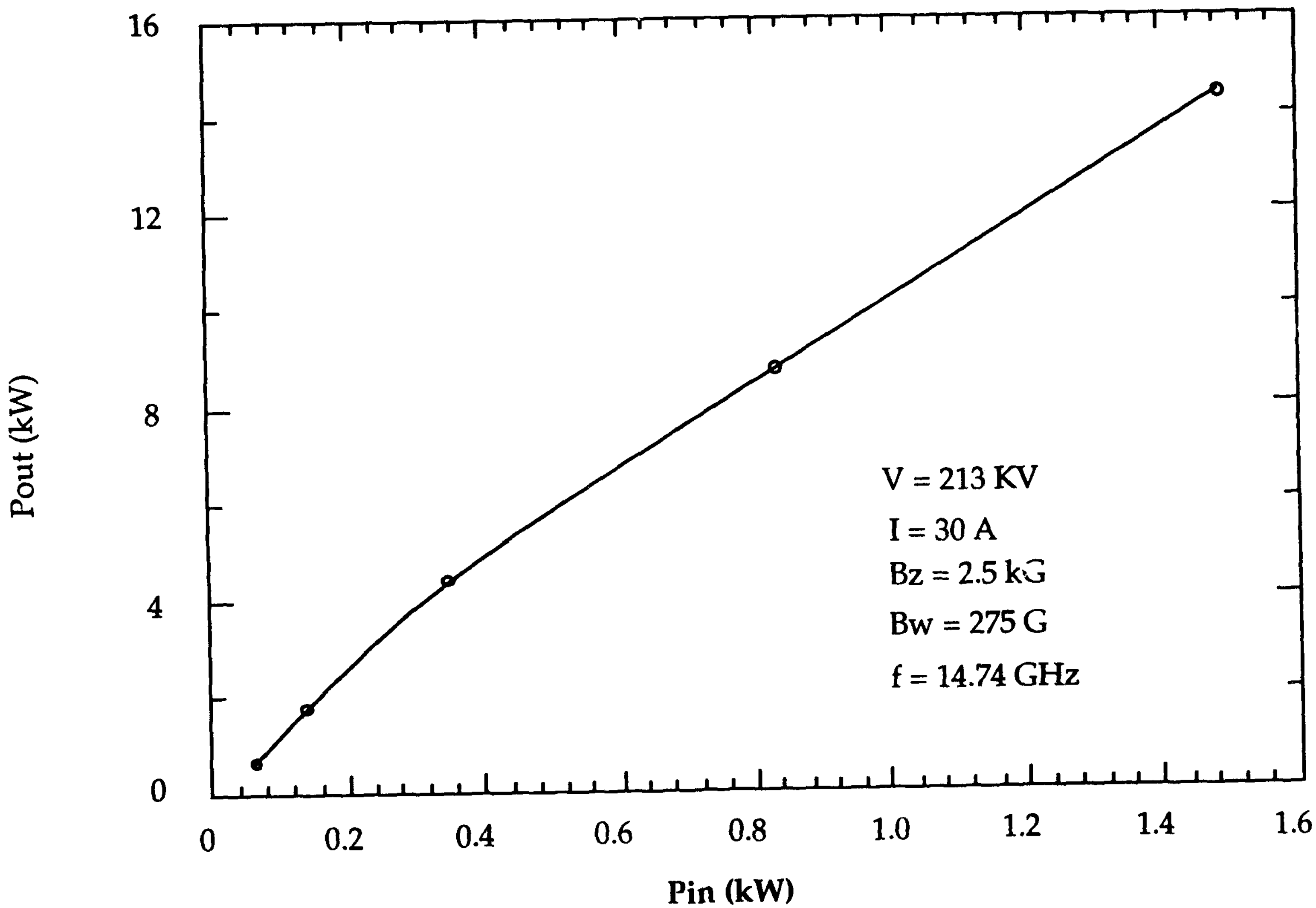


Figure 3.28. Output power dependence on input power.

The electromagnetic fields are represented in the form of a superposition of the vacuum waveguide modes. An arbitrary number of both TE and/or TM modes may be included in general, although only the TE_{11} mode is important in the present experiment. The microwave space-charge fields are approximated by the Gould-Trivelpiece modes of a fully-filled waveguide [23]. Although such a representation does not precisely correspond to the experimental configuration, it constitutes a reasonable approximation in the case of a grazing incidence interaction [6].

A comparison of the ga' ' as found in theory and experiment is shown in Figs. 3.29 and 3.30 as functions of frequency for the parameter sets V = 217 kV, I = 30 A, B_Z = 2.5 kG, B_W = 282 G, and V = 232 kV, I = 27 A, B_Z = 2.75 kG, and B_W = 331 G, respectively. Fig. 3.29 illustrates the gain curves found (1) in the experiment, (2) in simulation with collective Raman effects disabled, and (3) in simulation with complete collective effects. In practice, the collective effects may be disabled by removal of the space-charge waves from the formulation. It is evident from the figure that the average gain found by means of the collective Raman theory is in good agreement with experimental results. Comparison of the Compton and Raman simulations shows the importance of collective effects to the experiment, since the collective effect results in a decrease in the gain by more than a factor of two. It should be noted, however, that the shape of the experimental spectrum is suggestive of a double peaked spectrum, which is in closer agreement with the Compton simulation.

Comparisons between the simulation results and the measured gain indicate that an initial axial velocity spread in the neighborhood of 1% is required in the simulation for quantitative agreement with measurements. The initial velocity spread used in the simulation describes the condition of the beam prior to the entry into the wiggler. As mentioned earlier, the electron trajectory code used to design the electron gun and transport system predicts velocity spreads on the order of 0.26%, or less. Subsequent increases in the velocity spread due to transverse wiggler gradients are included self-consistently in the simulation. That a 1% velocity spread is required is most likely due to a discrepancy between the actual and simulated wiggler magnetic fields. For the simulation, the wiggler field was assumed to be that of an ideal bifilar helix of 2.54 cm period, with 12 periods in the uniform field regions, 5 periods in the

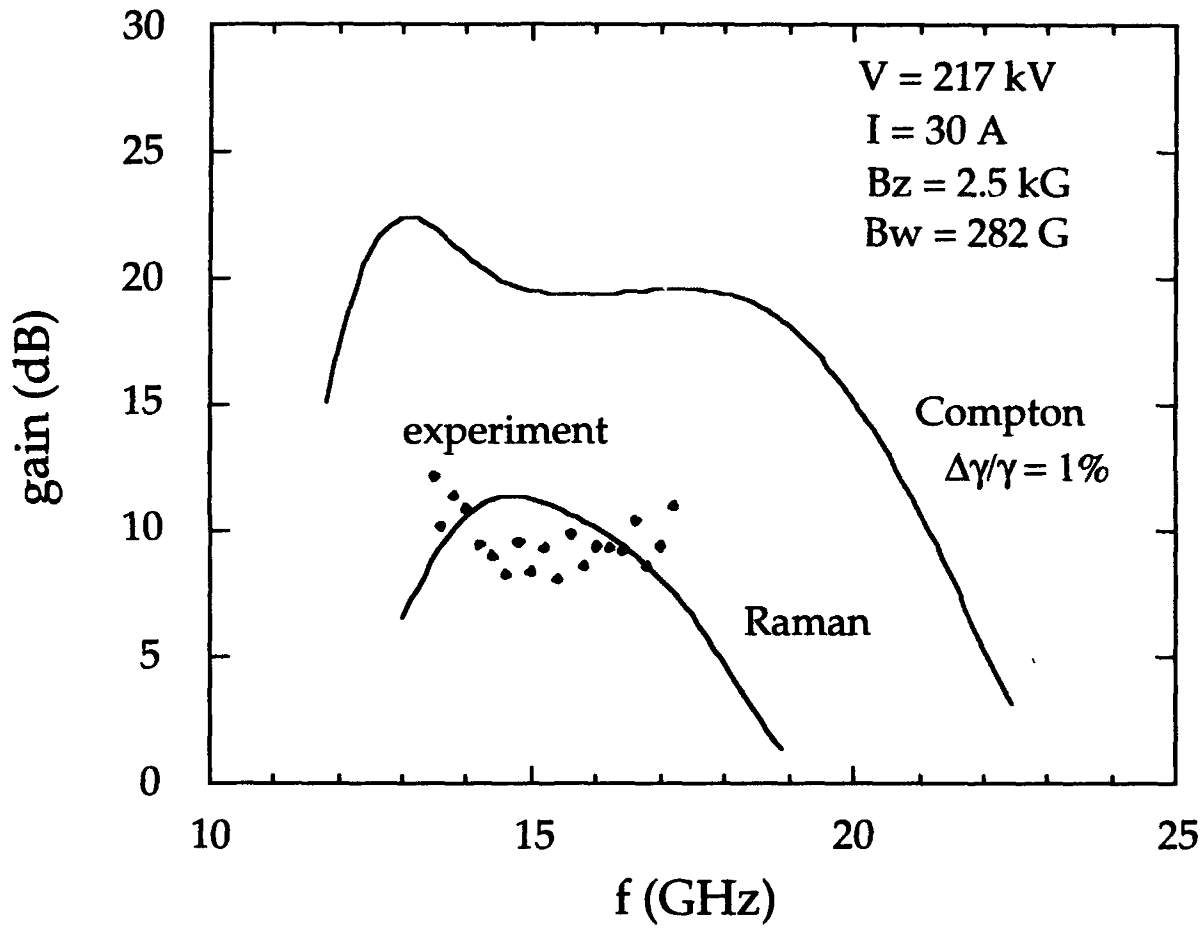


Figure 3.29. Gain comparison: experiment and theory, Compton and Raman regimes: V=21 kV, I=30 A, Bz=2.5 kG, Bw=282 G.

entrance taper, and 3 periods in the exit taper. Recall that it was not possible to measure the wiggler field until after all experimental data were acquired. The measured wiggler field profile departed significantly from the idealized model used in simulation.

The experimental results are, therefore, somewhat ambiguous as to the regime of operation. The shape of the experimental spectrum is similar to the Compton prediction, suggesting little collective interaction, while the average gain is close to the Raman regime prediction. The correlation between uncoupled dispersion curves and measured gain profiles for the measured parameter sets is also qualitatively suggestive of Compton regime operation. A double peaked distribution is measured for those cases where the beam line crosses the waveguide dispersion curve, and a flat or monotonic distribution is measured for those cases where the beam line is slightly below the dispersion curve. This type of behavior would not necessarily be expected for the coupled dispersion curves in the Raman regime. The much reduced gain in comparison with the Compton regime prediction might be a result of a much higher than expected velocity spread.

3.2.4 Laminar Beam Summary.

Amplifier operation of the NRL ubitron experiment has been achieved with a peak gain of 19 dB and an instantaneous bandwidth exceeding 25%. The measured peak gain per wavelength is 1.25 dB/λ at 13.5 GHz. The interaction has been identified by frequency, waveguide mode and amplification characteristics to be a fundamental wiggler harmonic ubitron/FEL interaction with the TE11 waveguide mode. Reasonable agreement has been obtained between measurements and theory concerning gain, bandwidth and general performance characteristics. In particular, higher values of gain per free-space wavelength have been achieved due to the combination of helical wiggler and circularly polarized waveguide mode. Performance was limited, in part, by an inability to operate reliably above 230 kV due to gun arcs. Use of the advanced electron gun should permit operation closer to the 250 kV design voltage, with correspondingly improved performance. While amplifier operation was generally stable, oscillation could be induced under some conditions. For example, oscillation was measured at the TE_{21} cutoff frequency, 17.8 GHz, for the parameters V = 235 kV, I = 35 A, B_z = 2 kG, and $B_w \sim$ 575 G.

SECTION 4
SUMMARY

The components for a single-stage/single-pass Ku band ubitron amplifier have been designed, constructed, and tested. Notable features of this device include a high quality electron beam and circularly polarized microwave and wiggler fields. Stable operation of the assembled amplifier has been achieved during initial testing using a modified SLAC klystron electron gun. In addition, amplifier and oscillator operation was measured using an unusual rippled beam parameter regime.

While operation at the design voltage of 250 kV was not possible due to gun arcs, amplifier operation was possible at lower voltages, albeit with reduced performance. The ubitron was only operated in the small signal regime; no large signal measurements (e.g. saturation) were attempted. In this regime, peak gains of 19 dB were measured with greater than 25% bandwidth. The general performance characteristics are in reasonable agreement with theoretical predictions, although there are discrepancies between the measured and theoretical gain vs. frequency profiles. More work is needed to determine whether this is a deficiency in the model or if operational characteristics are not adequately well known, e.g. details of the wiggler field profile.

During initial ubitron operation, an interesting amplifier/oscillator operational regime was discovered. Using a highly rippled beam, harmonic operation was found to be possible at voltages 10 - 30% of the nominal 250 kV required for fundamental mode operation, with only very small changes in the experimental configuration. In the amplifier mode, a peak gain of 24 dB was measured. A bandwidth of ~7% was measured at slightly lower peak gain. The peak ouput power was ~ 25 kW, corresponding to a 3% (unsaturated) efficiency. Oscillation was also detected and tentatively identified as a second harmonic interaction with the TE_{21} mode.

An investigation of ubitron amplifier performance limits, including performance enhancement through system parameter tapering, should be possible with operation at the design voltage using the advanced electron gun.

SECTION 5
APPENDICES

5.1 A - DATA ACQUISITION SYSTEM.

A basic data acquisition system has been developed for use in the ubitron experiment. The primary function of this system is the acquisition of component test and calibration data, not ubitron operational data acquisition. The system is typically used for thermistor/calorimeter calibration, network analyzer measurements, and magnetic field mapping, both DC and pulsed. A Digital Equipment Corporation PDP11/23 microcomputer with a Kinetic Systems 2920 bus adapter for a 3920 CAMAC crate controller and a National Instruments GPIB11V-2 GPIB interface card comprise the basis of the system. Programs are written in FORTRAN-77, using the RT-11 operating system.

Available CAMAC hardware includes:

- DSP Technology Model Optima-850 powered crate
- Kinetic Systems Model 3920 crate controller
- DSP Model 2032 32 channel digital voltmeter .
- DSP Model SMC-406/H stepping motor controller
- DSP Model DD-002 dataway display
- DSP Model WGR-241 word generator
- DSP Model RTC-018 real time clock

Available GPIB devices include:

- LeCroy Model 9400 2-channel digital oscilloscope
- Tektronix 7D20 digital oscilloscope plug-in
- Hewlett-Packard Model 8756A scalar network analyzer
- Wavetek Model 8502 2-channel peak power meter
- EIP Model 585 microwave pulse counter
- Keithley digital voltmeter Models: 174, 175, 197

Brief descriptions of CAMAC and GPIB acronyms follow. The CAMAC (Computer Automated Measurement And Control) standard (IEEE-583) originated in the high-energy physics community as a means of acquiring and processing large amounts of experimental data as well as controlling various devices. The GPIB (General Purpose Interface Bus) standard (IEEE-488) originated with Hewlett-Packard as a means for transferring data between meters, plotters, and other devices.

A CAMAC-based system typically consists of one or more 'crates' connected to one or more computers. The crate is a 'holder' for various interchangeable function modules from a variety of manufacturers and includes the necessary power supplies and a bus for communication between modules and computers. These modules, which can include both analog and digital input and output functions, must reside in the crate in order to function. This system is capable of both high-speed and high-volume data acquisition. In contrast, a GPIB-based system links various stand-alone instruments, such as digital multimeters, network analyzers, and digital oscilloscopes. A comprehensive computerized data acquisition and control system utilizes both interface standards in a complementary fashion.

At present, only calorimeter data is acquired by computer during the experiment, although a basic program has been developed to transfer waveforms from the LeCroy oscilloscopes to computer for further processing. A high-resolution magnetic field mapping system incorporating a stepper motor driven positioning slide, a three-axis Hall probe, and the basic data acquisition system were used extensively for solenoid and wiggler field measurements. This has been especially useful in measuring the rapid spatial variation of the transverse field components in the helical wiggler. Corrections are made in software for deviations from the nominal control current and for sensitivity variations due to temperature fluctuations. A schematic of the field mapping system is shown in Fig. 5.1. The major data acquisition programs developed under this program are listed below, including a brief description of use. Program listings follow.

BGET1	acquires 3-axis magnetic field data as function of z
HLX2	acquires pulsed 3-axis magnetic field data using LeCroy digital scope; waveform data is smoothed, with peak detection
BGTLCR	uses LeCroy for pulsed waveform acquisition
LCRY12	all channels acquired from LeCroy, including memories and functions
THERM	calorimeter/thermistor calibration
CALDAT	calorimeter data acquisition
NETWK	scalar network analyzer measurements
PKPWR5	pulsed microwave waveform acquisition using Wavetek peak power meter

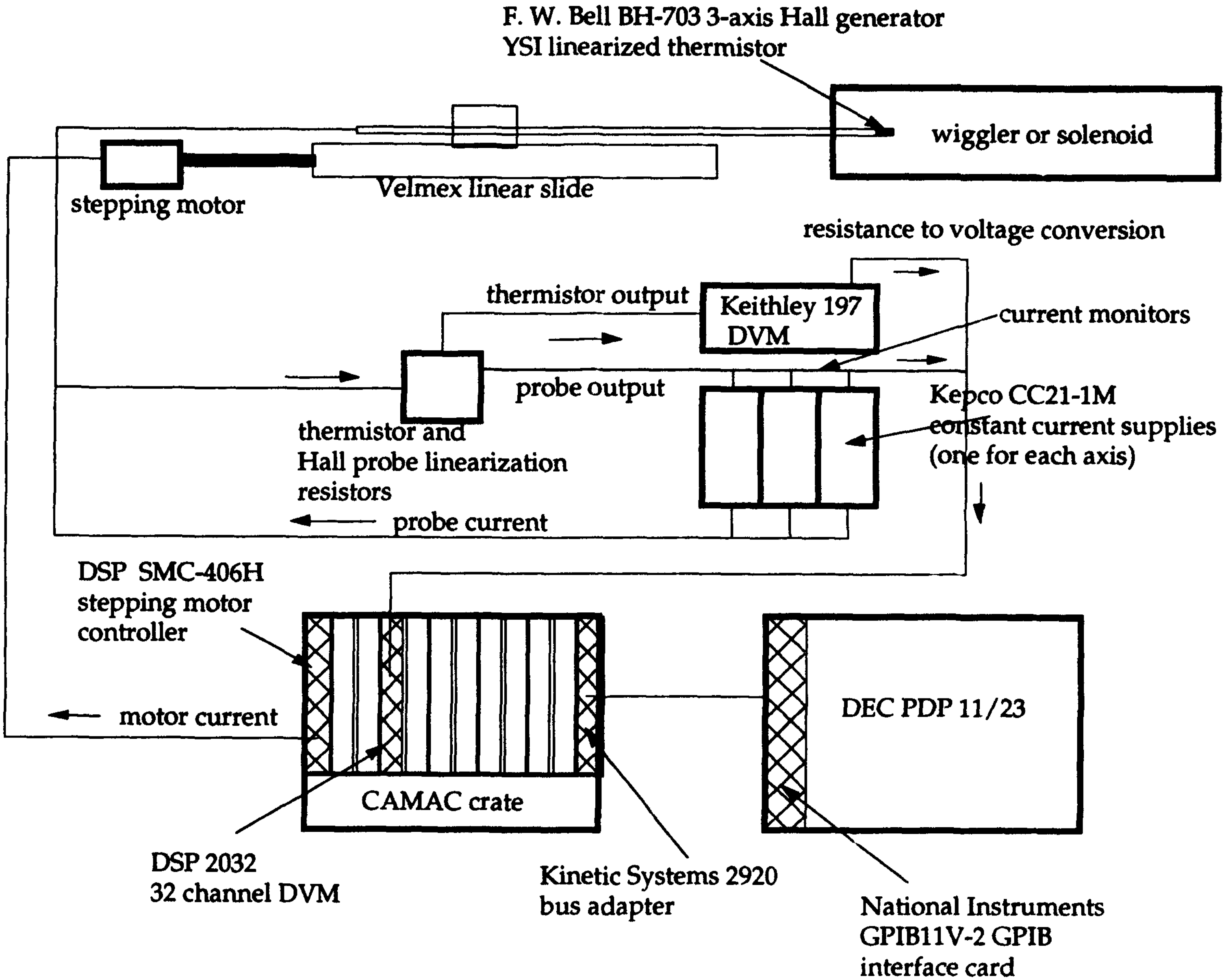


Figure 5.1. Schematic of magnetic field mapping system.

```
      PROGRAM  BGET1
      INTEGER Q,X,DATA(64),F16,F17
      INTEGER*4 J1,J2,J3
      LOGICAL*1 STRNG(8),ITAB
      CHARACTER*1 TAB
      REAL VDAT(32),ICXS,ICYS,ICZS,MTRSLD
      EQUIVALENCE (TAB,ITAB)
      COMMON/SCALE/VHXS,VHYS,VHZS,ICXS,ICYS,ICZS,MTRSLD
      COMMON/DVM/F16,F17,J1
      OPEN(UNIT=10,FILE='BGET.DAT',STATUS='NEW')
      ITAB=9
C---------------SCALE FACTORS-------------------------------------
      VHXS=6.733
      VHYS=6.766*0.932
      VHZS=6.776
      ICXS=1.0
      ICYS=1.0
      ICZS=1.0
      MTRSLD=4000.
C---------------------------------------------------------------------

C--------------INITIALIZATIONS-------------------------------------
C``````````````````````CRATE INITIALIZATION````````````````````````
C   STEPPER MOTOR CONTROLLER IN SLOT 2, DVM IN SLOT 5
      CALL INOFF
      CALL CRATEZ
      CALL CRATEC
      CALL CAMAC(5,0,9,0,0)
C```````````````````````````````````````````````````````````````````````

C----------------------------------------------------------------------

C----------------ENTER MEASUREMENT PARAMETERS------------------------
      TYPE *,'ENTER ZSTART,ZSTOP,STEP(ALL IN INCHES)'
      ACCEPT *, ZSTART,ZSTOP,STEP
      IF((ZSTOP-ZSTART).GE.0.) THEN
      IDIREC=1
      ELSE
      IDIREC=-1
      END IF
C    CONVERT 0 HRS, 0 MIN, X SECONDS TO INTEGER*4 VALUE
      TYPE *,'ENTER MEASUREMENT WAIT TIME IN SECONDS'
      ACCEPT *,JX
      JX=JX/2
      CALL JTIME(0,0,JX,0,J1)
      CALL IQSET(5)
      IF(IQSET(5).NE.0) STOP 'NOT ENOUGH FREE SPACE FOR QUEUE ELEMENTS'
C    SELECT TYPE OF DVM SCAN
  10  TYPE *,'ENTER 0 FOR SLOW SCAN, 2 FOR FAST SCAN'
      F16=2    !LET DEFAULT BE FAST SCAN
      ACCEPT *, F16
      IF(F16.NE.0.AND.F16.NE.2) GOTO 10
      WRITE(10,88)
  88  FORMAT(2X,'TIME',9X,'Z(in)',6X'Bx(kG)',6X,'By(kG)',6X,'Bz(kG)'
     >  ,7X,'T(C)')
C----------------------------------------------------------------------
```

```
C--------------------START SCAN--------------------------------------
      DO 20 Z=ZSTART,ZSTOP,STEP*IDIREC
      IF(Z.EQ.ZSTART) GO TO 15
      CALL MOTION(STEP,IDIREC,MOTERR)
      IF(MOTERR.EQ.0.OR.MOTERR.EQ.1) THEN
      CONTINUE
      ELSE
      WRITE(10,*)'FLAG CONDITION=',MOTERR
      STOP
      END IF

 15   CALL BDATA(STRNG,BX,BY,BZ,T)

      WRITE(10,98) (STRNG(K),K=1,8),TAB,Z,TAB,BX,TAB,BY,
     >  TAB,BZ,TAB,T
      TYPE 99,(STRNG(K),K=1,8),Z,BX,BY,BZ,T
  98  FORMAT (' ',8A1,5(A1,F11.4))
  99  FORMAT (' ',8A1,5F12.4)
!     PAUSE 'WAIT'
 20   CONTINUE
C--------------------END SCAN-------------------------------------

      END


      SUBROUTINE MOTION(STEP,IDIREC,MOTERR)
      INTEGER Q,X,DATA(64),F16,F17
      REAL VDAT(32),ICXS,ICYS,ICZS,MTRSLD
      COMMON/SCALE/VHXS,VHYS,VHZS,ICXS,ICYS,ICZS,MTRSLD
      COMMON/DVM/F16,F17
C  STEPPER MOTOR IN SLOT 2
C  CHECK STATUS REGISTER
C     CALL CAMAC(2,12,1,0,MOTERR)
C     IF(MOTERR.NE.0) RETURN
      ISTEPS=IFIX(STEP*MTRSLD)
      IF(IDIREC.EQ.1) ISTEPS=-(32767-ISTEPS)
      IF(ISTEPS.GT.32767.OR.ISTEPS.LT.-32767) THEN
      WRITE(10,*)'Z LIMITS EXCEEDED'
      RETURN
      END IF
C  ADD SOMETHING FOR MANUAL STOP DURING SCAN
      CALL CAMAC(2,0,16,0,ISTEPS)
 10   CALL CAMAC(2,12,1,0,MOTERR)
      IF(MOTERR.NE.0) GO TO 20
      GO TO 10
 20   RETURN
      END


      SUBROUTINE BDATA(STRNG,BX,BY,BZ,T)
      INTEGER Q,X,DATA(64),F16,F17
      INTEGER*4 J1,J2,J3
      LOGICAL*1 STRNG(8)
      REAL VDAT(32),ICXS,ICYS,ICZS,MTRSLD
      COMMON/SCALE/VHXS,VHYS,VHZS,ICXS,ICYS,ICZS,MTRSLD
      COMMON/DVM/F16,F17,J1
```

```
      EQUIVALENCE (DATA,VDAT)

C````````````````````DVM INITIALIZATION``````````````````````````

      F17=0         !START CHANNEL=0 (0-31 RANGE)
C    VX=25(ARRAY=26),VY=26(27),VZ=27(28)
C    IX=28(29),IY=29(30),IZ=30(31),T=31(32)
      CALL CAMAC(5,0,16,0,F16)
      CALL ITWAIT(J1)
      IF(ITWAIT(J1).NE.0) STOP 'NO QUEUE ELEMENT AVAILABLE'
      CALL TIME(STRNG)
      CALL CAMAC(5,0,17,0,F17)
      DO 10 I=1,32
      DO 10 J=1,2
 10   DATA(J+(I-1)*2)=0
C`````````````````````````````````````````````````````````````````

      DO 20 I=1,32
      DO 20 J=1,2
      CALL CAMAC(5,0,0,K,DATA(J+(I-1)*2))
      IF(X().NE.1.OR.Q().NE.1) WRITE(10,*)I,J,'X=',X(),' Q=',Q()
 20   CONTINUE
!     DO 30 I=1,32
! 30  WRITE(10,*) VDAT(I)

      CALL TEMP(VDAT(32),T)
C------------TEMPERATURE CORRECTIONS?----------------------------

      BX=VDAT(26)*1000./VHXS/VDAT(29)
      BY=VDAT(27)*1000./VHYS/VDAT(30)
      BZ=VDAT(28)*1000./VHZS/VDAT(31)
      RETURN
      END

      SUBROUTINE TEMP(VDAT32,T)
      R=ABS(VDAT32)*10000.
      T=(R-4593.39)/(-32.402)
      RETURN
      END
```

```
      PROGRAM  HLX2
      INTEGER Q,X,DATA(64),F16,F17
      INTEGER*4 J1,J2,J3
      LOGICAL*1 STRNG(8),ITAB
      CHARACTER*1 TAB
      REAL VDAT(32),ICXS,ICYS,ICZS,MTRSLD
      EQUIVALENCE (TAB,ITAB)
      COMMON/SCALE/VHXS,VHYS,VHZS,ICXS,ICYS,ICZS,MTRSLD
      COMMON/DVM/F16,F17,J1
      OPEN(UNIT=10,FILE='BGET.DAT',STATUS='NEW')
      ITAB=9
C---------------SCALE FACTORS------------------------------------
      VHXS=6.733
      VHYS=6.766*0.932
      VHZS=6.776
      ICXS=1.0
      ICYS=1.0
      ICZS=1.0
      MTRSLD=4000.
C-----------------------------------------------------------------

C--------------INITIALIZATIONS------------------------------------
C```````````````````````CRATE INITIALIZATION```````````````````````
C  STEPPER MOTOR CONTROLLER IN SLOT 2, DVM IN SLOT 5
      CALL INOFF
      CALL CRATEZ
      CALL CRATEC
      CALL CAMAC(5,0,9,0,0)
      J=IBUP(2,0)
      J=IBUP(0,1,'FOR1X',5)
      J=IBUP(0,1,'S8X',3)
C``````````````````````````````````````````````````````````````````

C------------------------------------------------------------------

C----------------ENTER MEASUREMENT PARAMETERS-----------------------
      TYPE *,'ENTER ZSTART,ZSTOP,STEP(ALL IN INCHES)'
      ACCEPT *, ZSTART,ZSTOP,STEP
      IF((ZSTOP-ZSTART).GE.0.) THEN
      IDIREC=1
      ELSE
      IDIREC=-1
      END IF
C   CONVERT 0 HRS, 0 MIN, X SECONDS TO INTEGER*4 VALUE
      TYPE *,'ENTER MEASUREMENT WAIT TIME IN SECONDS'
      ACCEPT *,JX
      JX=JX/2
      CALL JTIME(0,0,JX,0,J1)
      CALL IQSET(5)
      IF(IQSET(5).NE.0) STOP 'NOT ENOUGH FREE SPACE FOR QUEUE ELEMENTS'
C   SELECT TYPE OF DVM SCAN
   10 TYPE *,'ENTER 0 FOR SLOW SCAN, 2 FOR FAST SCAN'
      F16=2    !LET DEFAULT BE FAST SCAN
      ACCEPT *, F16
      IF(F16.NE.0.AND.F16.NE.2) GOTO 10
      WRITE(10,88)
   88 FORMAT(2X,'TIME',9X,'Z(in)',6X,'B(kG)')
```

```
C-------------------------------------------------------------------------

C--------------------START SCAN-------------------------------------------
      DO 20 Z=ZSTART,ZSTOP,STEP*IDIREC
      IF(Z.EQ.ZSTART) GO TO 15
      CALL MOTION(STEP,IDIREC,MOTERR)
      IF(MOTERR.EQ.0.OR.MOTERR.EQ.1) THEN
      CONTINUE
      ELSE
      WRITE(10,*)'FLAG CONDITION=',MOTERR
      STOP
      END IF

 15   CALL BDATA(STRNG,B1)

      WRITE(10,98) (STRNG(K),K=1,8),TAB,Z,TAB,B1
      TYPE 99,(STRNG(K),K=1,8),Z,B1
  98  FORMAT (' ',8A1,A1,F11.4,A1,E13.5)
  99  FORMAT (' ',8A1,F12.4,E14.5)
!     PAUSE 'WAIT'
 20   CONTINUE
C--------------------END SCAN---------------------------------------------

      END


      SUBROUTINE MOTION(STEP,IDIREC,MOTERR)
      INTEGER Q,X,DATA(64),F16,F17
      REAL VDAT(32),ICXS,ICYS,ICZS,MTRSLD
      COMMON/SCALE/VHXS,VHYS,VHZS,ICXS,ICYS,ICZS,MTRSLD
      COMMON/DVM/F16,F17
C  STEPPER MOTOR IN SLOT 2
C  CHECK STATUS REGISTER
C     CALL CAMAC(2,12,1,0,MOTERR)
C     IF(MOTERR.NE.0) RETURN
      ISTEPS=IFIX(STEP*MTRSLD)
      IF(IDIREC.EQ.1) ISTEPS=-(32767-ISTEPS)
      IF(ISTEPS.GT.32767.OR.ISTEPS.LT.-32767) THEN
      WRITE(10,*)'Z LIMITS EXCEEDED'
      RETURN
      END IF
C  ADD SOMETHING FOR MANUAL STOP DURING SCAN
      CALL CAMAC(2,0,16,0,ISTEPS)
 10   CALL CAMAC(2,12,1,0,MOTERR)
      IF(MOTERR.NE.0) GO TO 20
      GO TO 10
 20   RETURN
      END


      SUBROUTINE BDATA(STRNG,B1)
      BYTE V1(20),V11(12)
      INTEGER Q,X,DATA(64),F16,F17
      INTEGER*4 J1,J2,J3
      LOGICAL*1 STRNG(8)
      REAL VDAT(32),ICXS,ICYS,ICZS,MTRSLD
      COMMON/SCALE/VHXS,VHYS,VHZS,ICXS,ICYS,ICZS,MTRSLD
```

```
      COMMON/DVM/F16,F17,J1
      EQUIVALENCE (DATA,VDAT)

C`````````````````````DVM INITIALIZATION``````````````````````````


      F17=0        !START CHANNEL=0 (0-31 RANGE)
C     VX=25(ARRAY=26),VY=26(27),VZ=27(28)
C     IX=28(29),IY=29(30),IZ=30(31),T=31(32)
      CALL CAMAC(5,0,16,0,F16)
      CALL ITWAIT(J1)
      IF(ITWAIT(J1).NE.0) STOP 'NO QUEUE ELEMENT AVAILABLE'
      CALL TIME(STRNG)
      CALL CAMAC(5,0,17,0,F17)
      DO 10 I=1,32
      DO 10 J=1,2
 10   DATA(J+(I-1)*2)=0
C``````````````````````````````````````````````````````````````````

      DO 20 I=1,32
      DO 20 J=1,2
      CALL CAMAC(5,0,0,K,DATA(J+(I-1)*2))
      IF(X().NE.1.OR.Q().NE.1) WRITE(10,*)I,J,'X=',X(),' Q=',Q()
 20   CONTINUE
      J=IBUP(1,1,V1,18)
      DO 13 II=1,12
 13   V11(II)=V1(II+4)
      DECODE(12,100,V11) R1
 100  FORMAT(E13.5)
C     CALL TEMP(VDAT(32),T)
C------------TEMPERATURE CORRECTIONS?----------------------------

C     BX=VDAT(26)*1000./VHXS/VDAT(29)
C     BY=VDAT(27)*1000./VHYS/VDAT(30)
C     BZ=VDAT(28)*1000./VHZS/VDAT(31)
      RETURN
      END

      SUBROUTINE TEMP(VDAT32,T)
      R=ABS(VDAT32)*10000.
      T=(R-4593.39)/(-32.402)
      RETURN
      END
```

```
      PROGRAM  BGTLCR
C
C  PROGRAM COMBINES THE CAMAC AND GPIB CONTROLERS TO AQUIRE MAGNETIC
C    FIELD DATA.  CAMAC STEPS THE PROBE, AND GPIB CONTROLS THE LECROY
C    TO AQUIRE THE CURRENT AND FIELD VALUES.  PROGRAM ALSO USES BGTLE1.
        INTEGER Q,X,DATA(64),F16,F17
        INTEGER*4 J1,J2,J3
        LOGICAL*1 STRNG(8),ITAB
        CHARACTER*1 TAB
        CHARACTER*10 FILOUT
        REAL VDAT(32),ICXS,ICYS,ICZS,MTRSLD
        EQUIVALENCE (TAB,ITAB)
        COMMON/SCALE/VHYS,ICXS,ICYS,ICZS,MTRSLD
        COMMON/DVM/F16,F17,J1
        TYPE *,'ENTER OUTPUT ''FILE NAME'''
        ACCEPT *,FILOUT
        OPEN(11,FILE=FILOUT,STATUS='UNKNOWN')
        ITAB=9
C---------------SCALE FACTORS-------------------------------------
        VHXS=6.733
        VHYS=6.766*0.932
        VHZS=6.776
        ICXS=1.0
        ICYS=1.0
        ICZS=1.0
        MTRSLD=4000.
C------------------------------------------------------------------

C--------------INITIALIZATIONS-------------------------------------
C``````````````````````CRATE INITIALIZATION``````````````````````
C  STEPPER MOTOR CONTROLLER IN SLOT 2, DVM IN SLOT 5
        CALL INOFF
        CALL CRATEZ
        CALL CRATEC
C        CALL CAMAC(5,0,9,0,0)
C  PUT LECROY IN REMOTE MODE
        J=IBUP(2,0)
C       J=IBUP(9,2,0104,044,000,002,0043111,002)
C  SET UP THE FORMAT OF THE DATA THE LECROY SENDS OVER
        J=IBUP(0,2,'CBLS,60;CFMT,L,BYTE,UFIX',24)
C  UNMASK THE OPERATION COMPLETE BYTES TO SEND SRQ
        J=IBUP(0,2,'MASK 1,16',9)
        J=IBUP(0,2,'MASK 5,8',8)
C``````````````````````````````````````````````````````````````````

C------------------------------------------------------------------

C----------------ENTER MEASUREMENT PARAMETERS-----------------------
        TYPE *,'ENTER ZSTART,ZSTOP,STEP(ALL IN INCHES)'
        ACCEPT *, ZSTART,ZSTOP,STEP
        IF((ZSTOP-ZSTART).GE.0.) THEN
        IDIREC=1
        ELSE
        IDIREC=-1
        END IF
C   CONVERT 0 HRS, 0 MIN, X SECONDS TO INTEGER*4 VALUE
        TYPE *,'ENTER MEASUREMENT WAIT TIME IN SECONDS'
```

```
        ACCEPT *,JX
        JX=JX/2
        CALL JTIME(0,0,JX,0,J1)
        CALL IQSET(5)
        IF(IQSET(5).NE.0) STOP 'NOT ENOUGH FREE SPACE FOR QUEUE ELEMENTS'
C    SELECT TYPE OF DVM SCAN
C  10     TYPE *,'ENTER 0 FOR SLOW SCAN, 2 FOR FAST SCAN'
C         F16=2    !LET DEFAULT BE FAST SCAN
C         ACCEPT *, F16
C         IF(F16.NE.0.AND.F16.NE.2) GOTO 10
C      WRITE HEADER FOR OUTPUT FILE
        WRITE(11,88)
  88    FORMAT(2X,'TIME',9X,'Z(in)',6X'By(kG)',6X,'By1(kG)',7X,'T(C)')
C-------------------------------------------------------------------

C--------------------START SCAN-------------------------------------
        DO 20 Z=ZSTART,ZSTOP,STEP*IDIREC
        IF(Z.EQ.ZSTART) GO TO 15
        CALL MOTION(STEP,IDIREC,MOTERR)
        IF(MOTERR.EQ.0.OR.MOTERR.EQ.1) THEN
        CONTINUE
        ELSE
        WRITE(11,*)'FLAG CONDITION=',MOTERR
        STOP
        END IF

 15     CALL BDATA(STRNG,CURR,BY,CURREF,BYREF,T)

        WRITE(11,98) (STRNG(K),K=1,8),TAB,Z,TAB,CURR,TAB,BY,
     1   TAB,CURREF,TAB,BYREF,TAB,T
        TYPE 99,Z,TAB,CURR-CURREF,TAB,BY-BYREF
  98    FORMAT (' ',8A1,6(A1,F11.4))
  99    FORMAT (' ',F11.4,2(A1,F11.4))
!       PAUSE 'WAIT'
 20     CONTINUE
        J=IBUP(2,0)
        J=IBUP(5,0)
C--------------------END SCAN---------------------------------------

        END


        SUBROUTINE MOTION(STEP,IDIREC,MOTERR)
        INTEGER Q,X,DATA(64),F16,F17
        REAL VDAT(32),ICXS,ICYS,ICZS,MTRSLD
        COMMON/SCALE/VHYS,ICXS,ICYS,ICZS,MTRSLD
        COMMON/DVM/F16,F17
C   STEPPER MOTOR IN SLOT 2
C   CHECK STATUS REGISTER
C       CALL CAMAC(2,12,1,0,MOTERR)
C       IF(MOTERR.NE.0) RETURN
        ISTEPS=IFIX(STEP*MTRSLD)
        IF(IDIREC.EQ.1) ISTEPS=-(32767-ISTEPS)
        IF(ISTEPS.GT.32767.OR.ISTEPS.LT.-32767) THEN
        WRITE(11,*)'Z LIMITS EXCEEDED'
        RETURN
        END IF
```

```
C  ADD SOMETHING FOR MANUAL STOP DURING SCAN
        CALL CAMAC(2,0,16,0,ISTEPS)
 10     CALL CAMAC(2,12,1,0,MOTERR)
        IF(MOTERR.NE.0) GO TO 20
        GO TO 10
 20     RETURN
        END


        SUBROUTINE BDATA(STRNG,CURR,BY,CURREF,BYREF,T)
        BYTE V1(20),V4(20),V11(12),V42(11)
        INTEGER Q,X,DATA(64),F16,F17
        INTEGER*4 J1,J2,J3,J4
        LOGICAL*1 STRNG(8)
        REAL VDAT(32),ICXS,ICYS,ICZS,MTRSLD
        COMMON/SCALE/VHYS,ICXS,ICYS,ICZS,MTRSLD
        COMMON/DVM/F16,F17,J1
      COMMON/WVFRM/NSTP,PEAKA,PEAKFF,PEAKFE,NADDR,REFER,VREFER,BREFER
        EQUIVALENCE (DATA,VDAT)

C`````````````````````DVM INITIALIZATION`````````````````````````


C        F17=0        !START CHANNEL=0 (0-31 RANGE)
C    VX=25(ARRAY=26),VY=26(27),VZ=27(28)
C    IX=28(29),IY=29(30),IZ=30(31),T=31(32)
C        CALL CAMAC(5,0,16,0,F16)
        CALL ITWAIT(J1)
        IF(ITWAIT(J1).NE.0) STOP 'NO QUEUE ELEMENT AVAILABLE'
        CALL TIME(STRNG)
C        CALL CAMAC(5,0,17,0,F17)
C        DO 10 I=1,32
C        DO 10 J=1,2
C 10     DATA(J+(I-1)*2)=0
C`````````````````````````````````````````````````````````````````
C        DO 20 I=1,32
C        DO 20 J=1,2
C        CALL CAMAC(5,0,0,K,DATA(J+(I-1)*2))
C        IF(X().NE.1.OR.Q().NE.1) WRITE(11,*)I,J,'X=',X(),' Q=',Q()
C 20     CONTINUE
C        JJ=0
C SELECT FUNCTION E AND RESET THE AVERAGING AND THEN SELECT FCN F
C   AND REDEFINE IT TO BE SUMMED AVERAGING.
C
        J=IBUP(0,2,'SEL,FE',6)
        J=IBUP(0,2,'ARST',4)
        J=IBUP(0,2,'SEL,FF',6)
        J=IBUP(0,2,'RDF,AVG,SUMMED,,C2,500',22)
 46     ISRQ=1
C
C   WAIT FOR SERVICE REQUEST BEFORE CONTINUING. MUST WAIT UNTIL BOTH
C    AVERAGING OPERATIONS ARE DONE, I.E. TWO SERVICE REQUESTS.
C
        DO 47 ISRQ=1,10000
         J=IBUP(6,-1)
         IF(J.EQ.1)THEN
C
C  IF FIRST AVERAGING OPERATION IS DONE (SRQ=1) DO SERIAL POLL
```

```
         J=IBUP(6,2)
C
C  WAIT FOR 2 SECONDS TO ALLOW SECOND OPERATION TO FINISH
         CALL JTIME(0,0,1,0,J4)
         CALL ITWAIT(J4)
C
C  DO ANOTHER SERIAL POLL AND EXIT LOOP
         J=IBUP(6,2)
         GOTO 48
          ELSE
           IF(ISRQ.EQ.10000)GOTO 46
        ENDIF
 47     CONTINUE
C
C  COPY FUNCTION F (THE PROBE SIGNAL AVERAGE) INTO MEMORY C
 48    J=IBUP(0,2,'STO,FF,MC',9)
C
C  SELECT FUNC. F AND REDFINE IT TO DO SMOOTHING ON MEM. C
       J=IBUP(0,2,'SEL,FF',6)
       J=IBUP(0,2,'RDF,SMO,7,,MC',17)
C
C   CALL THE ROUTINES TO READ AND PROCESS THE LECROY DATA
        DO 14 II=1,2
        CALL DESREAD(II)
        CALL DATREAD(II)
14      CONTINUE
C        CALL TEMP(VDAT(32),T)
C------------TEMPERATURE CORRECTIONS?----------------------------

C        BY1=VDAT(27)*50.04
        CURR=PEAKFE
        BY=PEAKFF
        CURREF=VREFER
        BYREF=BREFER
        RETURN
        END

C        SUBROUTINE TEMP(VDAT32,T)
C        R=ABS(VDAT32)*10000.
C        T=(R-4593.39)/(-32.402)
C        RETURN
C        END
C
C       READS WAVEFORM DESCRIPTOR INFORMATION FROM THE LECROY
C       AND STORES IT IN AN ARRAY CALLED DESC, EXCEPT FOR THE
C       TRIGGER DELAY, AND THE ADDRESSES OF THE FIRST AND LAST
C       DATA POINTS WHICH MUST BE INTEGER*4 VARIABLES.
C
        SUBROUTINE DESREAD(II)
        COMMON/DSCRP/ GAIN,VGAIN,OFFSET
        COMMON/CHAR/ M,READER
        COMMON/BYT/ V1
        BYTE V1(200)
        CHARACTER*200 M
        CHARACTER*20 READER
        INTEGER DESC(17)
        INTEGER*4 TRGDEL,ADDR1,ADDR2
        READER='READ,FE.DE'
```

```
        IF(II.EQ.2)READER(6:7)='FF'
C
C       TELL LECROY TO READ THE DESCRIPTOR.
C
        J=IBUP(0,2,READER,10)
C
C       RECIEVE THE DESCRIPTOR AND STORE IT.
C
        DO 10 I=1,10
         J=IBUP(1,2,V1,200)
         WRITE(M,100,ERR=99) (V1(N),N=1,J-2)
         IF(I.EQ.1)READ(M,101,ERR=99)(DESC(K),K=1,14)
         IF(I.EQ.2)READ(M,102,ERR=99)DESC(15),TRGDEL,DESC(16)
     1          ,ADDR1,ADDR2
         IF(I.EQ.3)READ(M,103,ERR=99)DESC(17)
 10     CONTINUE
C
C   CONVERT DESCRIPTOR DATA INTO THE ACTUAL NUMBERS FOR GAIN, VERTICAL
C      OFFSET, AND VARIABLE VERTICAL GAIN.
C
        NGAIN=DESC(1)-21
        CALL CONVERT(NGAIN,GAIN)
        VGAIN=FLOAT(DESC(2))
        OFFSET=FLOAT(DESC(4))
 100    FORMAT(BZ,200(A,:))
 101    FORMAT(BZ,7X,2(I3.3),2(I6.6),10(I3.3))
 102    FORMAT(BZ,7X,I3,I12,3(I6.6))
 103    FORMAT(BZ,7X,I3)
        GO TO 200
 99     TYPE *,'I/O ERROR'
 200    RETURN
        END

        SUBROUTINE DATREAD(II)
C
C       READS THE WAVEFORM DATA FROM THE LECROY AND CALLS
C       WAVEFORM TO TRANSLATE IT.
C
        COMMON/CHAR/ M,READER
        COMMON/BYT/ V1
      COMMON/WVFRM/NSTP,PEAKA,PEAKFF,PEAKFE,NADDR,REFER,VREFER,BREFER
        BYTE V1(200)
        CHARACTER*200 M
        CHARACTER*5 CHADDR
        CHARACTER*20 READER
        NSTP=0
        PEAKA=0.0
        REFER=0.0
        NFLG=0
        READER(9:10)='DA'
        IF(II.EQ.1)READER(11:17)=',,,1700'
        IF(II.EQ.2)READER(11:20)=',,1000,500'
C
C  TELL LECROY TO READ DATA
        IF(II.EQ.1)J=IBUP(0,2,READER,17)
        IF(II.EQ.2)J=IBUP(0,2,READER,20)
C
C  RECEIVE DATA AND TRANSLATE
```

```
      DO 10 I=1,1800
       J=IBUP(1,2,V1,200)
       IF(J.EQ.4.AND.II.EQ.2)NFLG=1
       IF(J.EQ.4)GO TO 111
       WRITE(M,100,ERR=99)(V1(N),N=1,J-2)
       CALL WAVEFORM(M,I,II)
       IF(NSTP.EQ.20)GOTO 111
10    CONTINUE
100   FORMAT(BZ,200(A,:))
130   FORMAT(I5)
140   FORMAT(A5)
      GO TO 111
99    TYPE *,'I/O ERROR'
111   CONTINUE
      IF(NFLG.EQ.1) THEN
        READER(11:14)=',,1,'
        WRITE(CHADDR,130,ERR=99)NADDR
        READ(CHADDR,140,ERR=99)READER(15:)
        LENGT=LEN(READER)
        J=IBUP(0,2,READER,LENGT)
        J=IBUP(1,2,V1,200)
        WRITE(M,100,ERR=99) (V1(N),N=1,J-2)
        I3=3
        I4=1
        CALL WAVEFORM(M,I4,I3)
         ELSE
      ENDIF
      RETURN
      END


      SUBROUTINE WAVEFORM(M,I,II)
C**********************************************************
C     RECEIVES THE RAW DATA AND CONVERTS IT TO
C     VOLTS PER METER DATA TO GET THE ACTUAL WAVEFORM.
C       THEN THE DATA GETS PRINTED OUT IN COLUMN FORM
C       WITH THE TIME,CHANNEL1,CHANNEL2 IN THE COLUMNS
C       RESPECTIVELY.
C
C       GAIN=FIXED VERTICAL GAIN
C       VGAIN=VARIABLE VERTICAL GAIN
C       OFFSET=VERTICAL OFFSET
C       NGAIN AND NINTVL ARE THE POSITIONS IN THE GAIN
C       AND SAMPLING INTERVAL TABLES RESPECTIVELY WHICH
C       CORRESPOND TO THE RAW DESCRIPTOR DATA.
C*********************************************************
     COMMON/WVFRM/NSTP,PEAKA,PEAKFF,PEAKFE,NADDR,REFER,VREFER,BREFER
       COMMON/DSCRP/ GAIN,VGAIN,OFFSET
       BYTE ITAB
       CHARACTER*200 M
       CHARACTER*1 TAB
       DIMENSION DATA(17)
       EQUIVALENCE (TAB,ITAB)
       ITAB=9
C
C   READ WAVEFORM DATA FROM M
C
        J1=(I-1)*17
```

```
          READ(M,100) SDATA,(DATA(J),J=1,17)
C
C
C
C
C
        IF(II.EQ.3) GOTO 147
        DO 3 K=1,SDATA
          IF(II.EQ.1)THEN
            IF(I.EQ.1.AND.K.EQ.1)PEAK=DATA(1)-10.
            IF(DATA(K).GT.PEAK)NADDR=J1+K-1+1700
            IF(DATA(K).GT.PEAK)PEAK=DATA(K)
            IF(I.EQ.1.AND.K.EQ.1)REFER=DATA(K)
             ELSE
              IF(II.EQ.2)REFER=REFER+DATA(K)
           ENDIF
 3      CONTINUE
        IF(PEAK.EQ.PEAKA.AND.II.EQ.1)NSTP=NSTP+1
        IF(PEAK.GT.PEAKA.AND.II.EQ.1.OR.NADDR.LT.2000)NSTP=0
        IF(NSTP.EQ.20.OR.SDATA.LT.17.AND.II.EQ.1)THEN
          PEAKFE=GAIN*((PEAK-128.)/32.-(OFFSET-200.)/25.)
     1                    *200./(VGAIN+80.)
          VREFER=GAIN*((REFER-128.)/32.-(OFFSET-200.)/25.)
     1                    *200./(VGAIN+80.)
          ELSE
          IF(II.EQ.2.AND.(J1+K).GE.995)BREFER=GAIN*((REFER/1000.
     1       -128.)/32.-(OFFSET-200.)/25)*200./(VGAIN+80.)*1000.
         ENDIF
        PEAKA=PEAK
 147    IF(II.EQ.3)PEAKFF=GAIN*((DATA(1)-128.)/32.-(OFFSET-200.)
     1              /25.)*200./(VGAIN+80.)*1000.
C        IF(II.EQ.3)TYPE *,II,NADDR
 100    FORMAT(5X,F2.0,17(F3.0,:))
 102    FORMAT(1P1E11.4)
 103    FORMAT(1P1E11.4,A1,1P1E11.4,A1,1P1E11.4)
        RETURN
        END


        SUBROUTINE CONVERT(NGAIN,GAIN)
C
C  THIS SUBROUTINE READS FROM TABLES THE ACTUAL VALUES OF
C  THE GAIN AND THE SAMPLING INTERVAL.
C
        DIMENSION GNTBLE(10),TRVLTB(35)
        DATA GNTBLE/.005,.01,.02,.05,.1,.2,.5,1.,2.,5./
        DATA TRVLTB/.2E-9,.4E-9,.8E-9,0.,0.,1E-8,2E-8,4E-8,8E-8,
     1       2E-7,4E-7,8E-7,2E-6,4E-6,8E-6,2E-5,4E-5,8E-5,2E-4,
     2       4E-4,8E-4,2E-3,4E-3,8E-3,2E-2,4E-2,0.,0.,0.,1E-7,
     3       1E-6,1E-5,1E-4,1E-3,1E-2/
        GAIN=GNTBLE(NGAIN)
        RETURN
        END
```

```
      program  lcry12
c
c     reads channels 1 and 2 from the LeCroy oscilloscope and
c     prints them out in column form to be replotted with
c     Cricket graph.
c
      common/blk1/ dat1,dat2,datc,datd,date,datf,kfunc
      character*13 reader
      character*10 filout,filret
      type *,'enter file to store lecroy raw data "file name"'
      accept *,filret
      type *,'do you want to read functions?  yes=1 no=0'
      accept *,kfunc
      open(24,file=filret,status='unknown')
c  put lecroy in remote mode
      j=ibup(2,0)
c  set up the format of the data the lecroy sends over
      j=ibup(0,9,'cbls,60;cfmt,1,byte,ufix',24)
      j=ibup(0,2,'cbls,60;cfmt,1,byte,ufix',24)
      do 10 num=9,2,-7
c  read channel1 descriptor
      reader='read,c1.de,1 '
      call desread(reader,num)
c  read channel1 waveform data
      reader(9:10)='da'
      l=1
      call datread(reader,num,l)
c  read channel2 descriptor
      reader(7:13)='2.de,1 '
      call desread(reader,num)
c  read channel2 waveform data
      reader(9:10)='da'
      l=2
      call datread(reader,num,l)
c  read memoryc descriptor
      reader='read,mc.de,1 '
      call desread(reader,num)
c  read memoryc waveform data
      reader(9:10)='da'
      l=3
      call datread(reader,num,l)
c  read memoryd descriptor
      reader(7:13)='d.de,1 '
      call desread(reader,num)
c  read memoryd waveform data
      reader(9:10)='da'
      l=4
      call datread(reader,num,l)
c  read function descriptors and waveforms if called for
      if(kfunc.eq.1)then
       reader(6:13)='fe.de,1 '
       call desread(reader,num)
       reader(9:10)='da'
       l=5
       call datread(reader,num,l)
       reader(6:13)='ff.de,1 '
       call desread(reader,num)
```

```
        reader(9:10)='da'
        l=6
        call datread(reader,num,l)
         else
      endif
 10   continue
c   go to local
 111  j=ibup(5,9)
      j=ibup(5,2)
      end

c
c     reads waveform descriptor information from the lecroy
c     and stores it in an array called desc, except for the
c     trigger delay, and the addresses of the first and last
c     data points which must be integer*4 variables.
c
      subroutine desread(reader,num)
      common/dscrp/ desc
      common/char/ m
      common/byt/ v1
      byte v1(200)
      character*200 m
      character*13 reader
      integer desc(17)
      integer*4 trgdel,addr1,addr2
c
c     tell lecroy to read the descriptor.
c
      j=ibup(0,num,reader,10)
c
c     recieve the descriptor and store it.
c
      do 10 i=1,10
       j=ibup(1,num,v1,200)
       write(m,100,err=99) (v1(n),n=1,j-2)
       if(i.eq.1)read(m,101,err=99)(desc(k),k=1,14)
       if(i.eq.2)read(m,102,err=99)desc(15),trgdel,desc(16)
     1         ,addr1,addr2
       if(i.eq.3)read(m,103,err=99)desc(17)
c  write to external file to write back to lecroy later
       write(24,*)m(1:j-2)
 10   continue
 100  format(bz,200(a,:))
 101  format(bz,7x,2(i3.3),2(i6.6),10(i3.3))
 102  format(bz,7x,i3,i12,3(i6.6))
 103  format(bz,7x,i3)
      go to 200
 99   type *,'i/o error'
 200  return
      end

      subroutine datread(reader,num,l)
c
c     reads the waveform data from the lecroy and calls
c     waveform to translate it.
c
      common/dscrp/ desc
```

```
      common/char/ m
      common/byt/ v1
      common/blk1/ dat1,dat2,datc,datd,date,datf,kfunc
      byte v1(200)
      character*200 m
      character*13 reader
      integer desc(17)
c
c     decide how many data points to skip to read a
c     maximum of 500 data points. 1=read all, 2=skip
c     every other one, etc.
c
      itm=desc(8)
      irec=desc(10)
      reader(12:13)='1 '
      if(itm.eq.12.and.irec.eq.0)reader(12:13)='2 '
      if(itm.eq.13.and.irec.eq.0)reader(12:13)='4 '
      if(itm.eq.14)reader(12:13)='10'
      if(itm.eq.15)reader(12:13)='20'
      if(itm.eq.16)reader(12:13)='40'
      if(itm.ge.17)reader(12:13)='50'
      if(itm.eq.7)reader(12:13)='2 '
      if(itm.eq.8.and.irec.ne.0)reader(12:13)='5 '
      if(itm.eq.9.and.irec.ne.0)reader(12:13)='10'
      if(itm.eq.10.and.irec.ne.0)reader(12:13)='20'
      if(itm.ge.11.and.irec.ne.0)reader(12:13)='50'
      read(reader,150)nskip
c
c  tell lecroy to read data
      j=ibup(0,num,reader,13)
c
c  receive data
      do 10 i=1,50
       j=ibup(1,num,v1,200)
       if(j.eq.4)go to 111
       write(m,100,err=99)(v1(n),n=1,j-2)
c  write to external file to write back to lecroy later
       write(24,*)m(1:j-2)
 10   continue
 100  format(bz,200(a,:))
 150    format(bz,11x,i2)
      go to 111
 99   type *,'i/o error'
 111  write(24,*)'#I'
      return
      end
```

```
      PROGRAM  THERM
      INTEGER Q,X,DATA(64),F16,F17
      INTEGER*4 J1,J2,J3
      LOGICAL*1 STRNG(8)
      REAL VDAT(32)
      COMMON/DVM/F16,F17
      OPEN(UNIT=10,FILE='THERM.DAT',STATUS='NEW')

C--------------INITIALIZATIONS-------------------------------------
C``````````````````````CRATE INITIALIZATION````````````````````````
C  DVM IN SLOT 5
      CALL INOFF
      CALL CRATEZ
      CALL CRATEC
      CALL CAMAC(5,0,9,0,0)
C``````````````````````````````````````````````````````````````````
C   CONVERT 0 HRS,0 MIN, X SECONDS TO INTEGER*4 VALUE
      TYPE *,'ENTER DELTA T IN SECONDS'
      ACCEPT *,JX
      JX=JX/2.
      CALL JTIME(0,0,JX,0,J1)
      TYPE *,J1
C     CALL JJCVT(J1)
C     TYPE *,J1
      CALL IQSET(5)
      IF(IQSET(5).NE.0) STOP 'NOT ENOUGH FREE SPACE FOR QUEUE ELEMENTS'
      DO 10 I=1,100
      CALL RDATA(R1,R2,R3)
      CALL TIME(STRNG)
      TYPE 99,(STRNG(K),K=1,8),R1,R2,R3
      WRITE(10,99),(STRNG(K),K=1,8),R1,R2,R3
 99   FORMAT (8A1,3F12.4)
      CALL ITWAIT(J1)
      IF(ITWAIT(J1).NE.0) STOP 'NO QUEUE ELEMENT AVAILABLE'
      CALL IPOKE("44,"100.OR.IPEEK("44))
      ICHAR=ITTINR()
      IF(ICHAR.GT.0) GO TO 20
  10  CONTINUE

  20  END


      SUBROUTINE RDATA(R1,R2,R3)
      INTEGER Q,X,DATA(64),F16,F17
      REAL VDAT(32)
      COMMON/DVM/F16,F17
      EQUIVALENCE (DATA,VDAT)

C`````````````````````DVM INITIALIZATION`````````````````````````


      F16=0        !SLOW SCAN
      F17=0        !START CHANNEL=0 (0-31 RANGE)
C    R1(V)=25[ARRAY=26], R2(V)=31[32], R3(V)=27[28]
      CALL CAMAC(5,0,16,0,F16)
      CALL CAMAC(5,0,17,0,F17)
      DO 10 I=1,32
      DO 10 J=1,2
```

```
 10    DATA(J+(I-1)*2)=0
C``````````````````````````````````````````````````````````````````````

       DO 20 I=1,32
       DO 20 J=1,2
       CALL CAMAC(5,0,0,K,DATA(J+(I-1)*2))
       IF(X().NE.1.OR.Q().NE.1) THEN
       TYPE *,'X OR Q=1'
       END IF
 20    CONTINUE

       R1=ABS(VDAT(26))*10000.
       R2=ABS(VDAT(32))*10000.
       R3=ABS(VDAT(28))*10000.
       RETURN
       END
```

```
      PROGRAM  CALDAT
      INTEGER*4 J1,J2,J3
      LOGICAL*1 STRNG(8),ITAB
      CHARACTER*1 T
      EQUIVALENCE(T,ITAB)
      OPEN(UNIT=10,FILE='CALDAT.DAT',STATUS='NEW')
      ITAB=9

C````````````````````INITIALIZATIONS````````````````````````````
      J=IBUP(2,0)
      J=IBUP(0,1,'F2R3X',5)    !THERMISTOR 4 (192)
      J=IBUP(0,1,'S8X',3)      !SET 192 F3 FILT,2RD/S
      J=IBUP(0,4,'R3X',3)      !THERMISTOR 5 (197)
      B4=-17.0912
      B5=-16.887
      RI4=0.0
      RI5=0.0
C```````````````````````````````````````````````````````````````

C   CONVERT 0 HRS,0 MIN, X SECONDS TO INTEGER*4 VALUE
      TYPE *,'ENTER DELTA T IN SECONDS'
      ACCEPT *,JX
      JX=JX/2.
      CALL JTIME(0,0,JX,0,J1)
      CALL IQSET(5)
      IF(IQSET(5).NE.0) STOP 'NOT ENOUGH FREE SPACE FOR QUEUE ELEMENTS'
      T1=SECNDS(0.)
C
      TYPE *,'ENTER MAX NO. DATA POINTS'
      ACCEPT *,IDATAP
      WRITE(10,*) 'DELTAT(S)        R4        R5       DELTA_T (C)'
      TYPE *,     'DELTAT(S)        R4        R5       DELTA_T (C)'

      DO 10 I=1,IDATAP
      CALL RDATA(R1,R2)
C  INITIAL RESISTANCE VALUES.......................................
      IF(I.EQ.1) THEN
      RI4=R1
      RI5=R2
      WRITE(10,*) '>>> RI4=',RI4,'    RI5=',RI5,'<<<'
      TYPE *,' RI4=',RI4,'    RI5=',RI5
      END IF
C...............................................................

C  CALCULATE TEMPERATURE DIFFERENCE AND ELAPSED TIME
      DLTAT=(R2-RI5)/B5-(R1-RI4)/B4
      DELTAT=SECNDS(T1)
      TYPE 99, DELTAT,T,R1,T,R2,T,DLTAT
      WRITE(10,99) DELTAT,T,R1,T,R2,T,DLTAT
 99   FORMAT (' ',F6.1,2(A1,F8.2),A1,F8.4)

C  WAIT FOR DELTA T SECONDS BEFORE TAKING NEW DATA
      CALL ITWAIT(J1)
      IF(ITWAIT(J1).NE.0) STOP 'NO QUEUE ELEMENT AVAILABLE'

C  CHECK FOR KEYBOARD ENTRY TO TERMINATE DATA ACQUISTION
      CALL IPOKE("44,"100.OR.IPEEK("44))
```

```
      ICHAR=ITTINR()
      IF(ICHAR.GT.0) GO TO 20

 10   CONTINUE

 20   TYPE *,'ENTER METER FLOW RATE (ML/MIN)'
      ACCEPT *,FLOW
      FLOW=(FLOW+33.78)/0.9655
      WRITE(10,*)'CORRECTED FLOW= ',FLOW,'ML/MIN'
      TYPE *,     'CORRECTED FLOW= ',FLOW,'ML/MIN'
      END


      SUBROUTINE RDATA(R1,R2)
      BYTE V1(20),V4(20),V11(12),V42(11)
      J=IBUP(1,1,V1,18)
      J=IBUP(1,4,V4,17)
      DO 10 I=1,12
10    V11(I)=V1(I+4)
      DO 20 I=1,11
20    V42(I)=V4(I+4)
      DECODE(12,100,V11) R1
      DECODE(11,100,V42) R2
100   FORMAT(E13.5)
      RETURN
      END
```

```
      program  netwrk
c   *********************************************
c     program for data aquisition on the network
c     analyzer (model hp 8756A). Reads channel 1
c     and/or channel 2 and the start and stop
c     frequencies from the sweeper.
c   *********************************************
      byte v1(300),itab
      character*1 tab
      character*8 m(401)
      character*8 mm
      character*13 strt,stop
      character*10 ot1fle,ot2fle
      character*2 ch
      character*75 headc1,headc2,sbhdc1,sbhdc2
      equivalence (tab,itab)
c   decide which channels to read
      type *,'enter 1-ch1, 2-ch2, 3-ch1 and ch2'
      accept *,kflg
c
c   input the output file names and the descriptive header
c   for each channel that is to be read.
c
      if(kflg.eq.2)go to 75
      type *,'enter channel 1 output "file name"'
      accept *,ot1fle
      open(10,file=ot1fle,status='unknown')
      type *,'enter "ch1 header"'
      accept *,headc1
      type *,'enter more "ch1 header"'
      accept *,sbhdc1
      write(10,150)headc1,sbhdc1
      if(kflg.eq.1)go to 76
 75   type *,'enter channel 2 output "file name"'
      accept *,ot2fle
      open(20,file=ot2fle,status='unknown')
      type *,'enter "ch2 header"'
      accept *,headc2
      type *,'enter more "ch2 header"'
      accept *,sbhdc2
      write(20,150)headc2,sbhdc2
 76   itab=9
c   set default channel to channel 1
      ch='c1'
c   clear all devices
      j=ibup(2,0)
c   initialize device table in ibup (probably not necessary)
      j=ibup(9,3,0120,060,000,002,012,002)
      j=ibup(9,6,0121,061,000,002,012,002)
c   choose ASCII format for output instead of binary
      j=ibup(0,3,'fd0',3)
c   loop to read data
      do 1 nn=1,2
c   if kflg=2 skip channel 1 and do channel 2
      if (kflg.eq.2) nn=2
      if (nn.eq.2) ch='c2'
c   tell network analyzer which channel
```

```
      j=ibup(0,3,ch,2)
c  output memory (calibration signal)
      j=ibup(0,3,'om;',3)
c  read memory from analyzer
      do 10 i=1,400
      j=ibup(1,3,v1,8)
      write(m(i),100,err=99)(v1(n),n=1,7)
 10   continue
c  read last point of memory
      j=ibup(1,3,v1,255)
      write(m(401),100,err=99)(v1(n),n=1,7)
c  assign passthrough device for network analyzer
c    in this case the sweeper
      j=ibup(0,3,'pt19;',5)
c  tell sweeper to output start and stop frequencies
      j=ibup(0,6,'opfa',4)
      type *,'into passthrough'
      j=ibup(1,6,strt,13)
      type *,'sweep',strt
      j=ibup(0,6,'opfb',4)
      type *,'through'
      j=ibup(1,6,stop,13)
      type *,stop
      read(strt,101)fstrt
      read(stop,101)fstop
c  normalize start and stop to Ghz
      fstrt=fstrt/1.e9
      fstop=fstop/1.e9
      fstep=(fstop-fstrt)/400
      freq=fstrt
c  choose input A for chan 1, input B for channel 2
      if (nn.eq.1) j=ibup(0,3,'ia',2)
      if (nn.eq.2) j=ibup(0,3,'ib',2)
      type *,'enter 1 to continue *********'
      accept *,kcont
c  tell analyzer to display measurement of chosen channel
      j=ibup(0,3,'me',2)
c  tell analyzer to output the data of the chosen channel
      j=ibup(0,3,'od;',3)
c  read the data of the chosen channel and write to the output
c  file specified for that channel, also write the memory and
c  data - memory.
      do 20 i=1,400
      j=ibup(1,3,v1,8)
      write(mm,100,err=99)(v1(n),n=1,7)
      read(m(i),102,err=99)cal
      read(mm,102,err=99)sig
      relsig=sig-cal
      if (nn.eq.1) write(10,200)freq,tab,cal,tab,sig,tab,relsig
      if (nn.eq.2) write(20,200)freq,tab,cal,tab,sig,tab,relsig
      freq=freq+fstep
 20   continue
c  do last data point
      j=ibup(1,3,v1,255)
      write(mm,100,err=99)(v1(n),n=1,7)
      read(m(401),102,err=99)cal
      read(mm,102,err=99)sig
      relsig=sig-cal
```

```
      if (nn.eq.1)write(10,200) freq,tab,cal,tab,sig,tab,relsig
      if (nn.eq.2)write(20,200) freq,tab,cal,tab,sig,tab,relsig
      if (kflg.eq.1) nn=2
 1    continue
 100  format(8(a,:))
 101  format(e12.5)
 102  format(f7.3)
 200  format(f7.4,a1,f7.3,a1,f7.3,a1,f7.3)
 150  format(a75/a75//2x,'freq',3x,'cali-',2x,'signal',3x,'signal'/
     1      1x,'(Ghz)',3x,'brate',10x,'-cal.'/)
      go to 111
  99  type *,'encode error'
c  go back to local
 111  j=gpib(14)
      end
```

```
      program  pkpwr5
      byte itab
      character*1 res,tab
      character*2 chan
      character*3 avel
      character*4 form,resol
      character*5 ofseta,ofsetb,reader
      character*6 freq1
      character*7 aver
      character*8 rep
      character*9 offa,offb,tm,powera(500)
      character*10 filout,freq2
      character*11 repr
      character*13 cursa,cursb
      character*38 freq
      character*80 power
      equivalence (tab,itab)
      itab=9
      type *,'enter desired "output file"'
      accept *, filout
      open(unit=15,file=filout,status='unknown')
      j=ibup(9,7,0105,045,0,0,0,2)
      j=ibup(0,7,'datf',4)
      aver='avpk   '
      reader='pkp '
      offa='offa     '
      offb='offb     '
      cursa(1:4)='cdla'
      cursb(1:4)='cdlb'
      freq2(1:4)='freq'
      ofseta='0'
      ofsetb='0'
      offa(5:9)=ofseta
      offb(5:9)=ofsetb
      write(15,*)'offset A=',ofseta,'dB'
      write(15,*)'offset B=',ofsetb,'dB'
      j=ibup(0,7,offa,9)
      j=ibup(0,7,offb,9)
c 2   type *, 'enter "averaging number"'
c     accept *, avel
c     aver(5:7)=avel
c     j=ibup(0,7,aver,5)
c     type *, 'do you want "a", or "b", or "ab"'
c     accept *, chan
c     if(chan.eq.'a'.or.chan.eq.'b')reader(4:5)=chan
c     if(chan.eq.'ab')reader(3:4)=chan
c     j=ibup(0,7,reader,4)
      form='watt'
      j=ibup(0,7,form,4)
      type *,'enter start time (in microseconds)'
      accept *, tme1
      type *,'enter end time (in us)'
      accept *, tme2
      type *,'enter number of points'
      accept *,npoint
      j=ibup(0,7,'updt',4)
      j=ibup(0,7,'pkpa',4)
```

```
2     do 10 ii=1,npoint
      time=tme1+(ii-1)*(tme2-tme1)/npoint
      write(tm,100) time
      cursa(5:13)=tm
      j=ibup(0,7,cursa,13)
      j=ibup(0,7,'updn',4)
      j=ibup(1,7,power,80)
      powera(ii)=power(5:13)
      if(jflg.eq.1)write(5,*)'after read from meter'
11      if(jflg.eq.1)power='0.0000'
 10   continue
      j=ibup(0,7,'pkpb',4)
      do 20 ii=1,npoint
      time=tme1+(ii-1)*(tme2-tme1)/npoint
      write(tm,100) time
      cursb(5:13)=tm
      j=ibup(0,7,cursb,13)
      j=ibup(0,7,'updn',4)
      j=ibup(1,7,power,80)
      write(15,*)tm,tab,powera(ii),tab,power(5:13)
      write(5,*)time,tab,powera(ii),tab,power(5:13)
20    continue
      type *, 'do you really want to do this again? (0=no 1=yes)'
      accept *,kflg
      if(kflg.eq.1)goto 2
      j=ibup(5,7)
100   format(f9.2)
      stop
      end
```

5.2 B - EXTERNAL SYSTEMS.

Details of construction and operation of each major component have been presented in earlier sections of this report. This appendix will outline, very briefly, the configuration of various external systems required for ubitron operation. These systems are the electron beam modulator, diagnostics/control, complete microwave circuit, vacuum, and coolant.

Electron Beam Modulator

The power source for the electron beam is a 'line type modulator' built by Beta Development Corporation. The pulse supplied to the cathode has a peak power rating of 62 MW (250A @ 250 kV), a duration of 1 μs, and a maximum repetition rate of 100 pps. Complete modulator specifications are listed in Table 5.1. A block diagram for the modulator is shown in Fig. 5.2, where 'klystron' refers to the ubitron electron gun.

Since the ubitron is designed to operate at a beam current of 30-100 A, a ballast resistor was added in parallel with the electron gun in order to present a matched 1 kΩ impedance to the modulator. The actual 'resistor' is comprised of a series - parallel arrangement, manufactured by OhmWeve Company, dissipating ~12 kW in oil, with resistance taps at 1.313, 1.456, 1.496, 1.609, 1.662, and 1.826 kΩ (see Fig. 5.3). A second current transformer was also added to monitor electron gun current, as shown in Fig. 5.4.

Table 5.1 Modulator Specifications.

Voltage	250 kV (max.)
Current	250 A
Power	62 MW (max.)
Width (@99% level)	1.0 μs (min.)
Rise time (1 to 99%)	1.0 μs (max.)
Fall time (99 to 1%)	1.0 μs (max.)
Flatness	± 1.0% (max.)
Jitter	± 10 ns (max.)
Amplitude drift (@ ± 5% line voltage variation)	

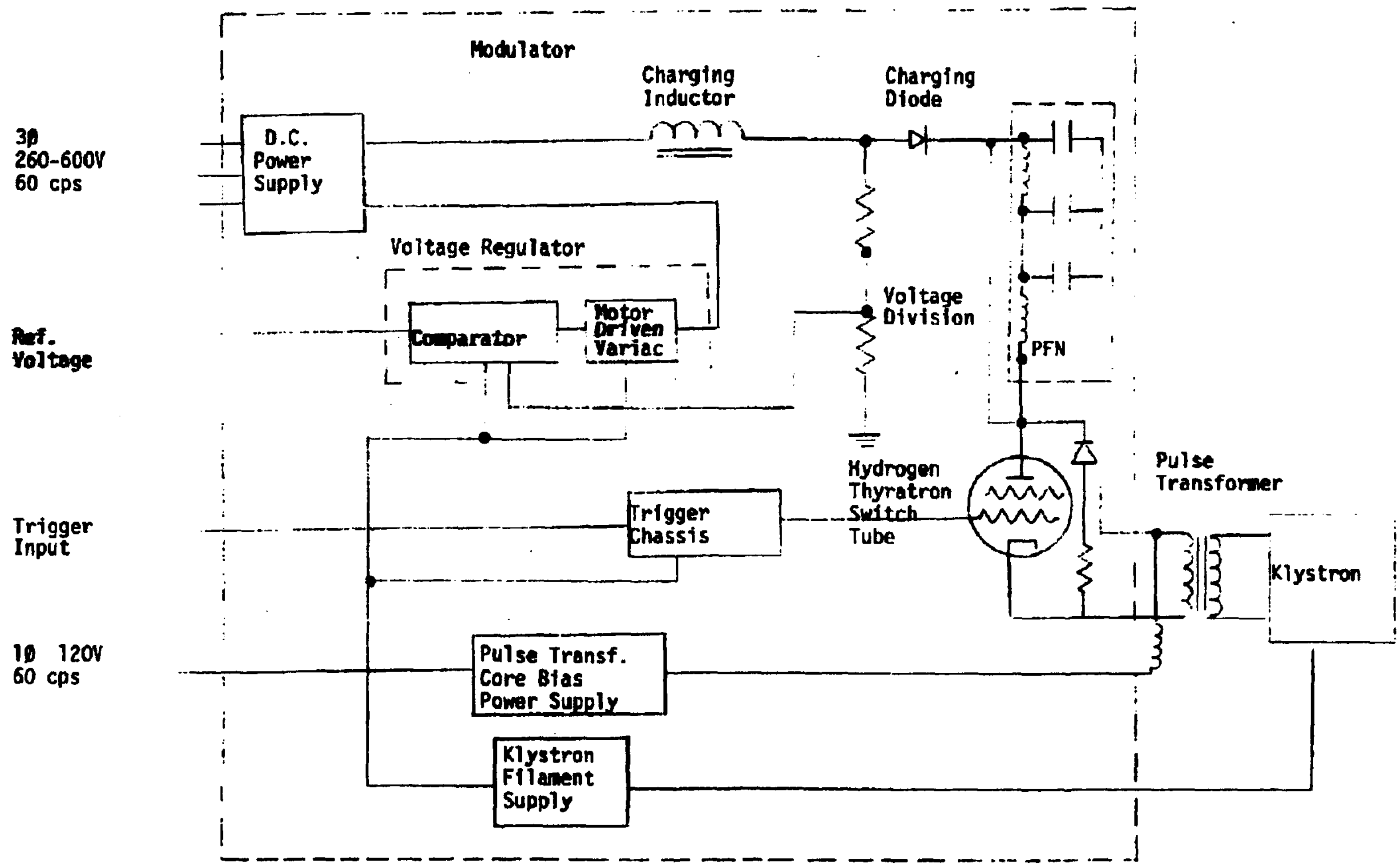


Figure 5.2. Modulator block diagram.

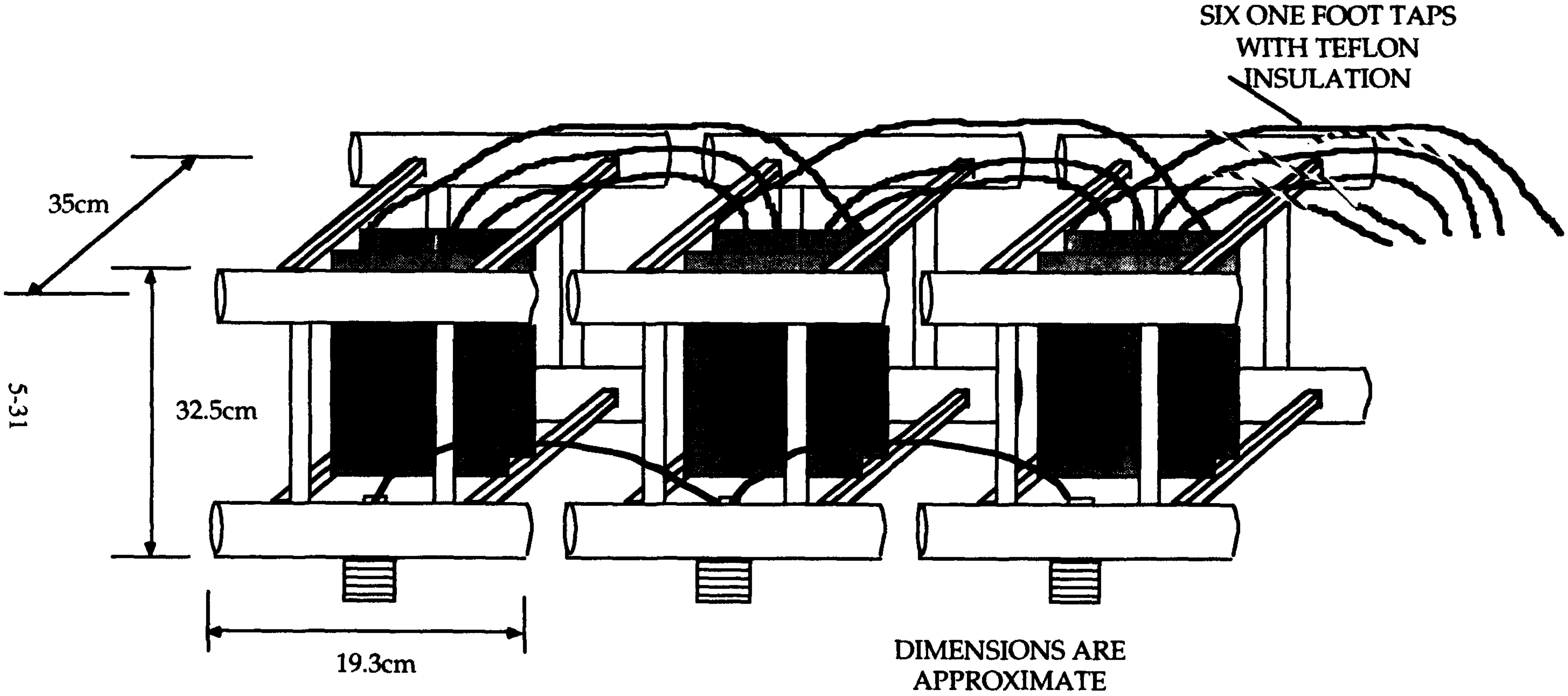


Figure 5.3. Modulator ballast resistor configuration.

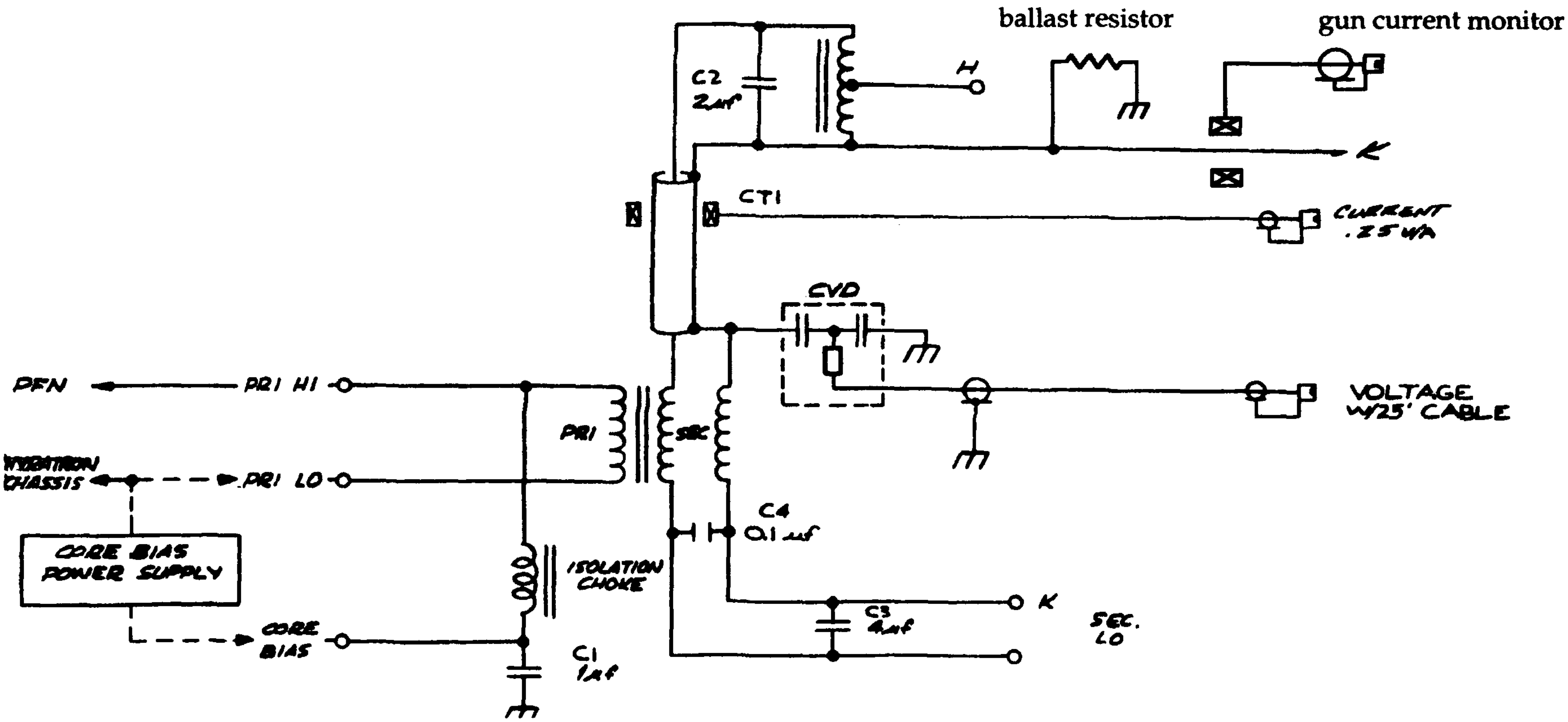


Figure 5.4. Modulator high voltage section (in oil tank).

Diagnostics/Control.

The diagnostic/control system for the ubitron is shown in Fig. 5.5. The various symbols, circle, triangle, and rectangle within an equipment box, e.g. helix power supply, indicate whether the particular item is recorded, checked, or controlled, respectively. Cable identification numbers are placed within each diamond, whose thickness indicates whether the information is send only, receive only, or send and receive.

Microwave Circuit.

Operation of the input coupler, including phase splitting circuitry, and the output coupler have been discussed in Secs. 2.5 and 2.6, respectively. A block diagram of the complete microwave circuit is shown in Fig. 5.6. Details of the complete input circuit are shown in Fig. 5.7, including all 'plumbing.' An HP 8690B/8695B sweeper with output power of 33 - 46 mW is the source driving the intermediate amplifier, either a Hughes (10-W) or Varian (20-W) TWTA. The intermediate amplifier, in turn, drives the Hughes/Litton high power amplifier (and modulator), with nominal 10-kW output power, which is the ubitron driver.

One final note concerning the microwave circuit. Due to the many Conflat joints in the ubitron, actual circuit performance is dependent upon careful alignment during assembly. The tube 'transmission loss,' measuring relative power transmitted through the output coupler after two separate tube assemblies, is shown in Fig. 5.8. Although considerable care was taken to insure alignment of the various components, the result of joint mismatches is evident.

Vacuum System.

After bakeout, the ubitron was intended to be operated with (4) Varian 0.2 l/s miniature ion pumps as maintainence pumps. While these pumps could maintain vacuum in the 10^{-8} Torr range with the beam propagating through to the collector, they were not able to keep up with the gas load during ubitron operation. This gas load resulted from beam lost to the vacuum chamber wall (waveguide) in the wiggler section during the rise and fall of the beam pulse.

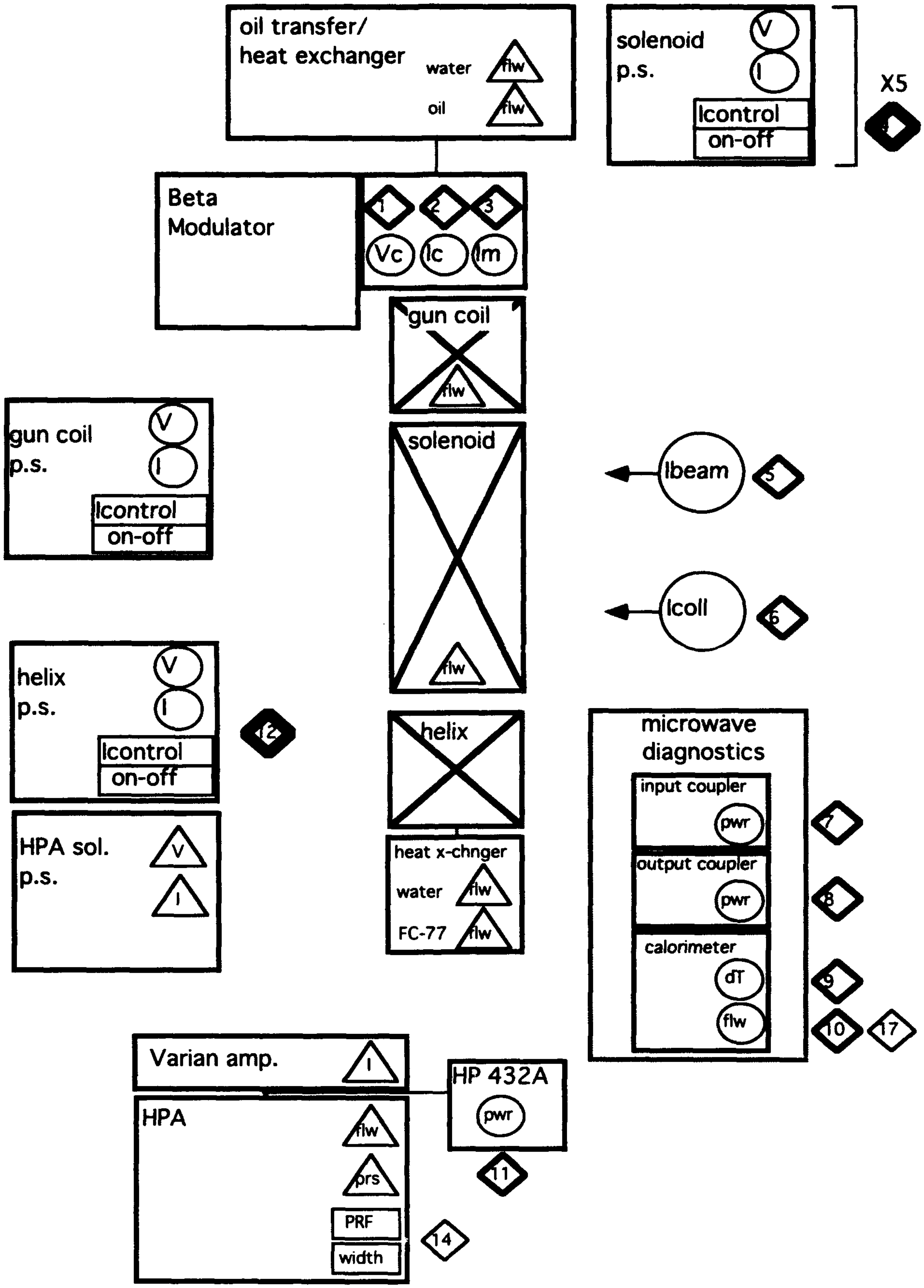
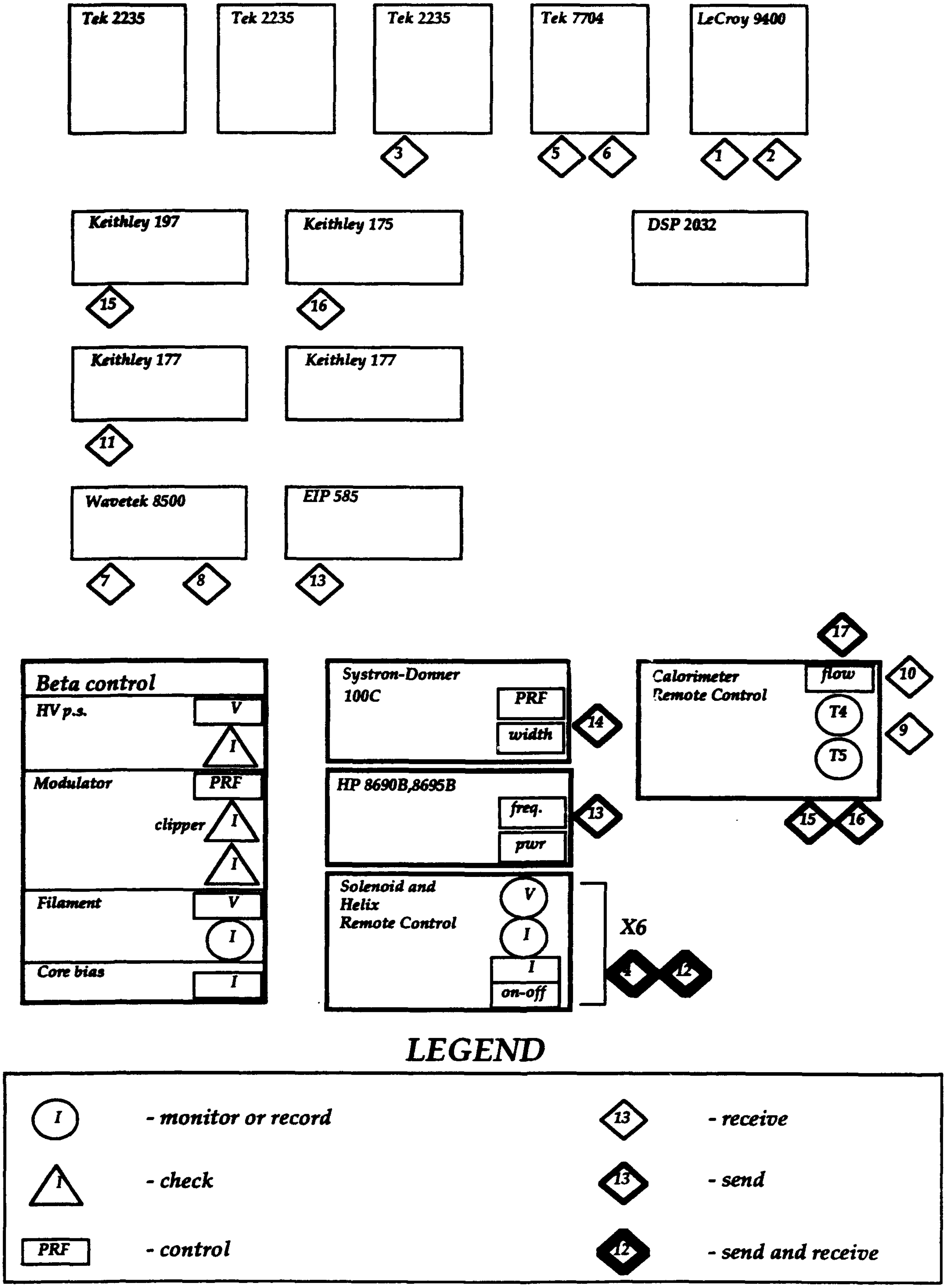


Figure 5.5. Ubitron diagnostics/control systems.



Tek 2235
Tek 2235
Tek 2235
Tek 7704
LeCroy 9400
3
5
6
1
2
Keithley 197
Keithley 175
DSP 2032
15
16
Keithley 177
Keithley 177
11
Wavetek 8500
EIP 585
7
8
13
17
Beta control
HV p.s.
V
I
Modulator
PRF
clipper
I
I
Filament
V
I
Core bias
I
Systron-Donner 100C
PRF
width
14
HP 8690B,8695B
freq.
pwr
13
Solenoid and Helix Remote Control
V
I
I
on-off
X6
4
12
Calorimeter Remote Control
flow
T4
T5
10
9
15
16
LEGEND
I
- monitor or record
I
- check
PRF
- control
13
- receive
13
- send
12
- send and receive

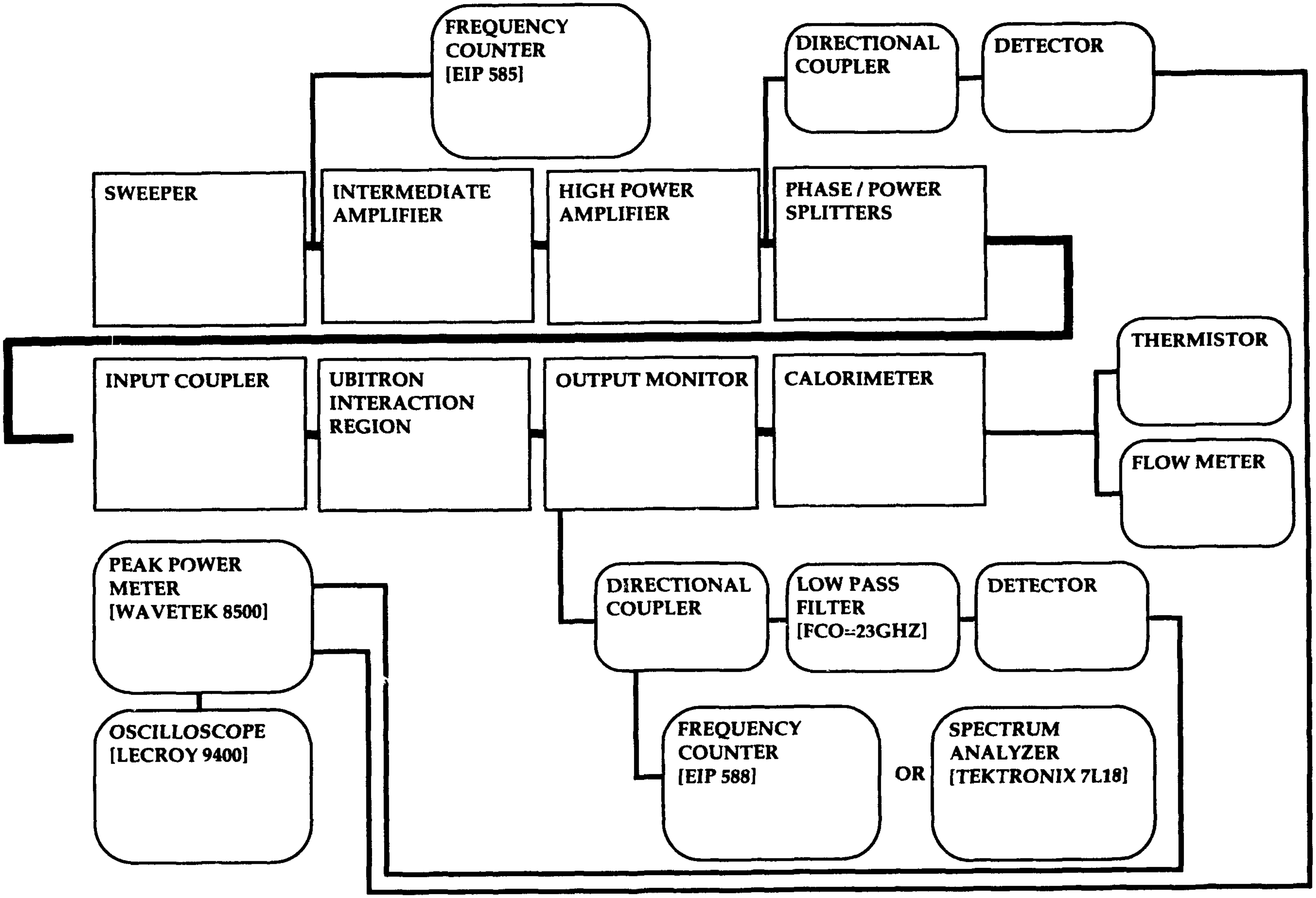


Figure 5.6. Block diagram of complete ubitron microwave circuit.

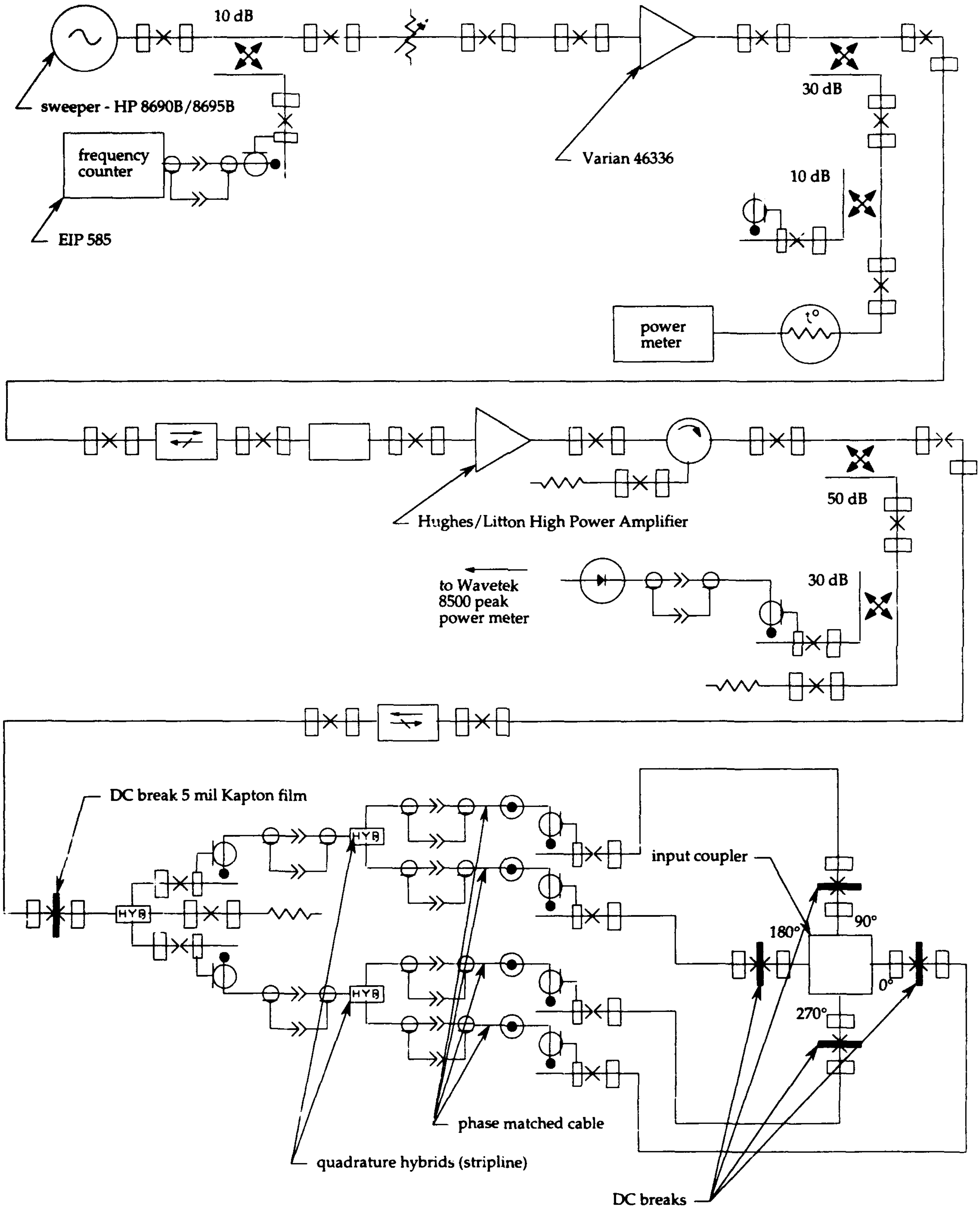


Figure 5.7. Complete ubitron microwave input circuit.

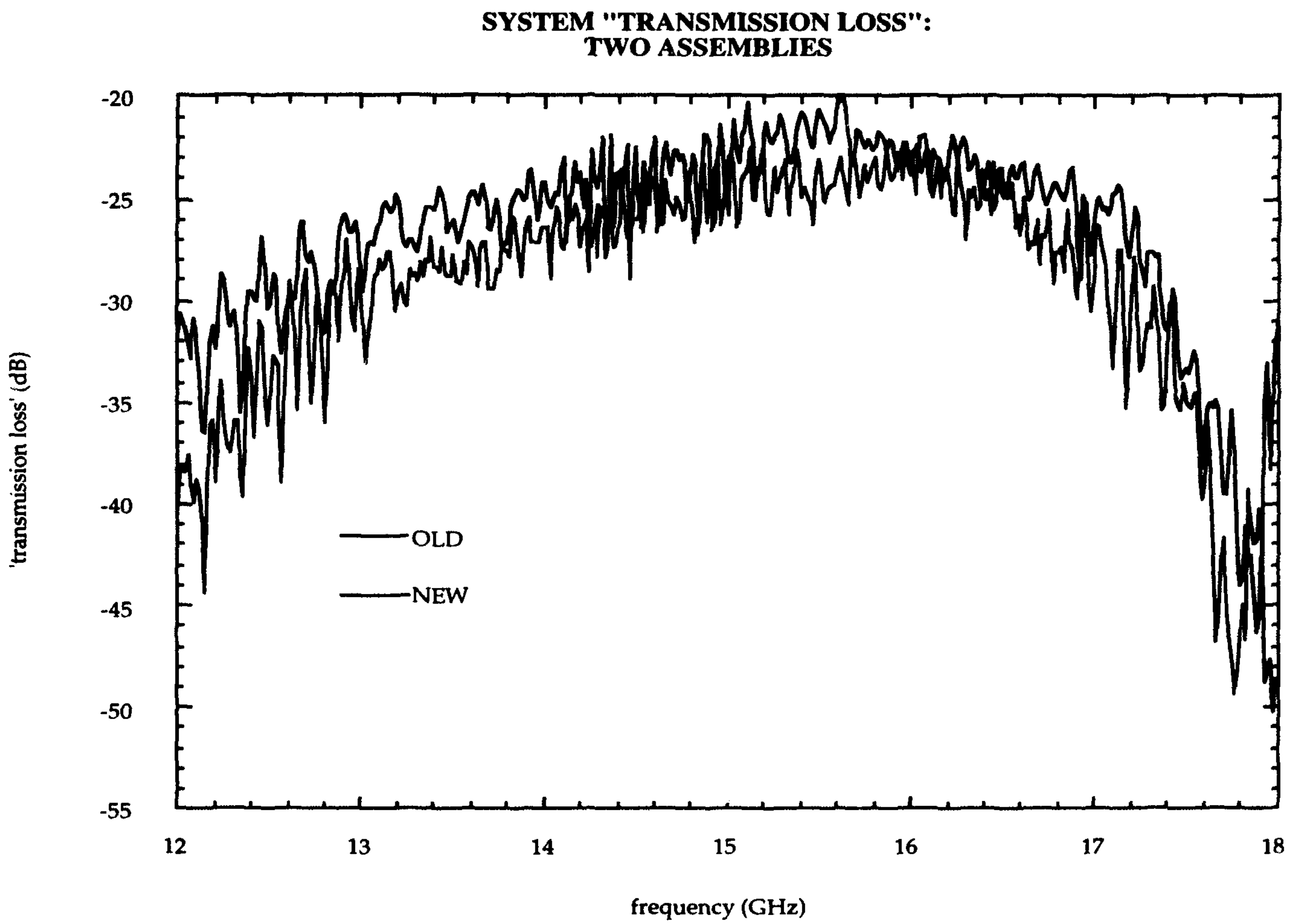


Figure 5.8. System "transmission loss" after two assemblies of ubitron.

To provide sufficient pumping speed during ubitron operation, a battery powered 8 l/s Varian ion pump was added behind the electron gun inside the high-voltage oil tank. This pump draws 10 - 50 μA at typical operating pressures and greater than 1 mA at pressures greater than 10^{-2} Torr. Since the pump must float at gun potential (250 kV), a rechargeable battery pack powering a high voltage DC - DC converter also floats at gun potential. The output of this power supply is nominally 1 mA @ 3V, unregulated, for 12 V, ~ 420 mA in. Two series connected Panasonic LCR6V1.3P gel cell batteries, 6 V and 1.3 AHr make up the battery pack. The circuit diagram is shown in Fig. 5.9. Up to 7 hours of battery operation are possible under normal conditions. However, the DC-DC converter can fail if the gun arcs, since there is no protection circuitry.

Coolant Systems.

Since this is a high power, rep-rated experiment, several coolant systems are required to maintain equipment or components at safe temperatures. There are four coolant systems: two tap water systems - one for the beam collector and scraper, and a second for the main solenoid coils, and two chilled, water-based systems. One chilled water system pumps Coolanol 25 (manufactured by Monsanto) through the large solenoid coil which is used with the advanced gun. The Coolanol is cooled using chilled water and a heat exchanger. A similar system pumps cooled FC-77 (manufactured by 3M) through the wiggler. A schematic diagram of all coolant systems is shown in Fig. 5.10.

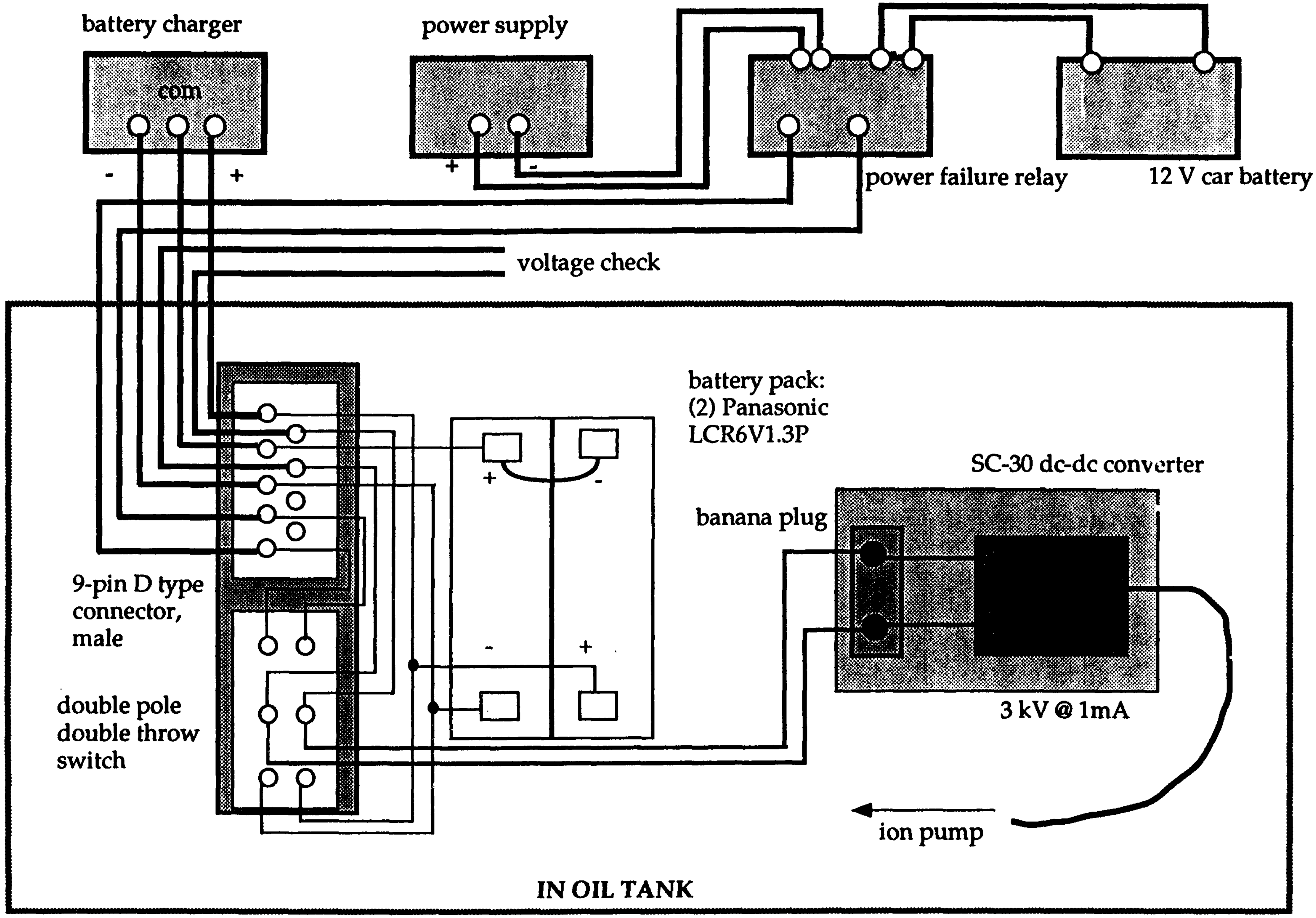


Figure 5.9. Circuit of battery powered ion pump power supply.

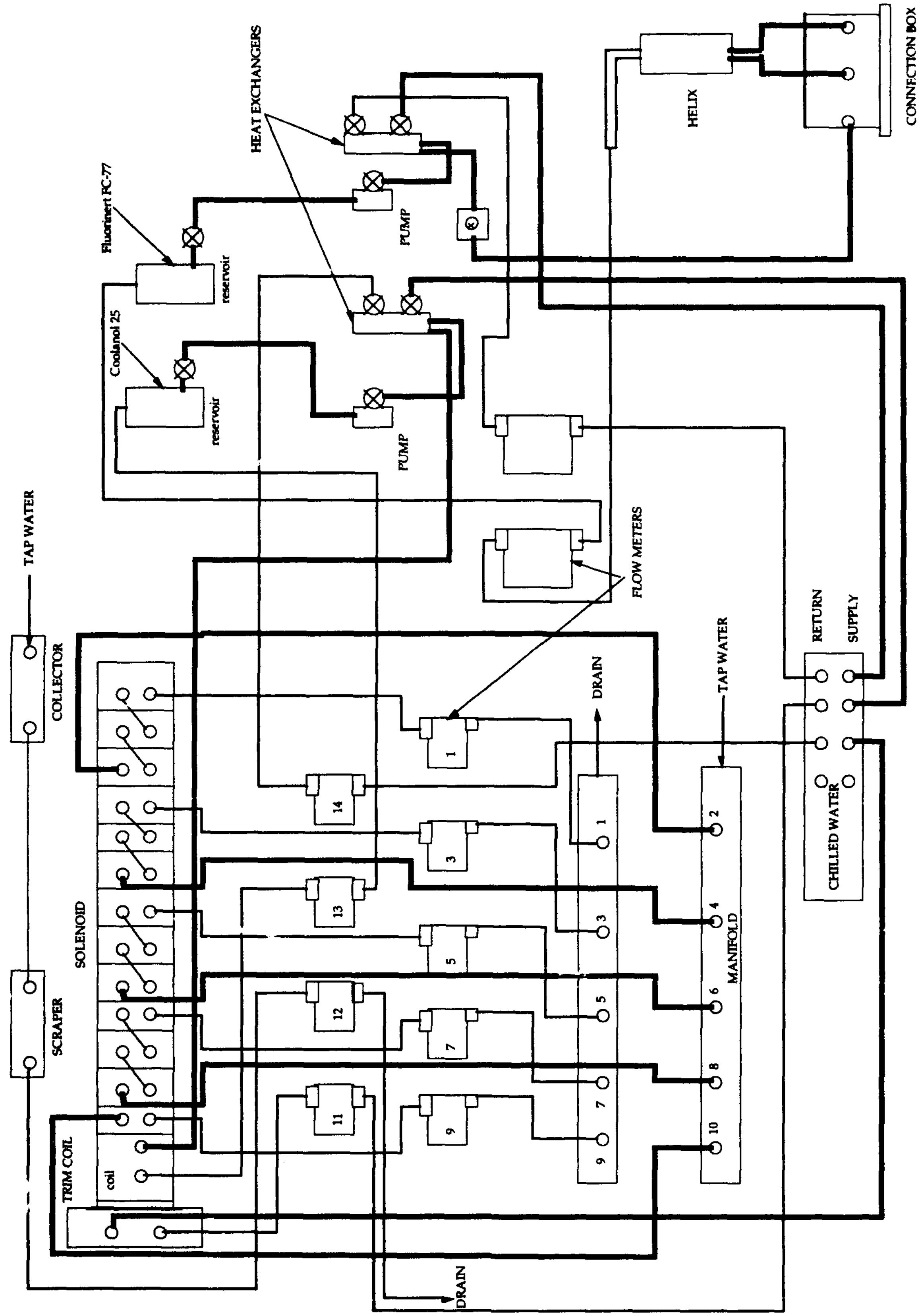


Figure 5.10. Schematic of ubitron coolant systems.

5.3 C - COMPUTER CODES.

In addition to design codes widely used in the electron device community, such as SCRIBE (a version of EGUN [10]) for electron trajectory calculations and POISSON for 2-D magnetics, several specialized computer codes were instrumental to the design of the ubitron. These codes were primarily concerned with magnetic field computation. Codes used for the output coupler design are discussed in Mission Research Corporation Report MRC/WDC-R-131. Brief code descriptions follow.

SMPLXGLTS - This code uses a Simplex minimization routine to determine the coil currents that will generate an axial magnetic field profile, the 'goal curve,' or any specified axial magnetic field profile. The goal curve is specified by a table of (z, Bz) pairs. The input data is a table of computed (by POISSON) magnetic field values at each of the same z values in the goal curve table, (z,Bz1,Bz2,Bz3,...), where Bz1 refers to magnetic field generated by coils connected to the first power supply, and so on. The code determines the coefficients x[i] that minimize the functional $(Bzd(z) - Bzc(z))^2$, where $Bzc(z) = x[1]*Bz1(z) + x[2]*Bz2(z) + \dots$.

INVHLX - FORTRAN program due to Robert Jackson that inverts the formula for the on-axis transverse magnetic field of an ideal bifilar helix to determine the winding profile that would generate a specified field taper. This is discussed in slightly more detail in Section 2.4.

RIBHLX and RIBTST2 - RIBHLX calculates the interior field of an infinite bifilar helix based on the series formulation of Park, Baird, Smith, and Hirshfield [24]. A finite rectangular cross-section conductor is approximated by four current filaments at each vertex. RIBTST2 uses RIBLHLX as a subroutine to calculate the average field across a rectangular area. This is used to calculate the ideal response of a rectangular cross-section Hall probe to the field generated by an ideal bifilar helix.

PRESCRIBE - This is a menu-driven interactive preprocessor designed to prepare input for the design codes, SCRIBE and POISSON. It is written in VAX

FORTRAN and requires a VT-100 terminal. The code is discussed in more detail in Mission Research Corporation Report MRC/WDC-R-247 [Ref. 25].

Code listings for SMPLXGLTS, INVHLX, and RIBTST2 follow.

```
program simplex;  {curve fitting with the simplex algorithm}

{by Marco Caceci, with help form William Caceris.  1983}
{Chem. Dept. Florida State University Tallahassee FL32306}
{no copy-right. sssd floppy disk copies on request}

{see Nelder J.A. & R. Mead, Computer J. 7, 308 (1965) and }
{L.A. Yarbo & S.N. Deming, Anal. chim. Acta 74,391 (1974) }

        const
                date = '1/11/90';
                memo = ' GOAL CURVE FIT/new solenoid configuration';


                m = 5; {number of parameters to fit}
                nvpp = 2; {total number of vars per data point}
                n = m + 1; {the value should be m+1}
                mnp = 350; {maximum number of data points}

                alfa = 1.0; {reflection coefficient, >0}
                beta = 0.5; {contraction coefficient, 0to1}
                gamma = 2.0; {expansion coefficient, >1}

                lw = 6; {width of line in data fields+1}
                page = 12;
                root2 = 1.414214;
                c = 3.0e8;
                pi = 3.1415926536;

        type
                vector = array[1..n] of real;
                datarow = array[1..nvpp] of real;
                index = 0..225;
        var
                done: boolean; {convergence}
                i, j: index;
                h, l: array[1..n] of index;{number high/low paramts}
                np,             {number of data points}
                maxiter,        {max number of iterations}
                niter: integer;  {number of iterations}
                next,           {new vertex to be tested}
                center,         {center of hyperplane described by all vertices}
{               of the simplex excluding the worst}
                mean, error, maxerr,        {maximum error accepted}
                p, q,           {to compute first simplex}
                step: vector;  {input starting steps}
                simp: array[1..n] of vector;   {the simplex}
                data: array[1..mnp] of datarow; {the data}
{**********************************************************************}
                din, dout, din1, fname: text;          {input, output}
                BP1, BP2, BP3, BP4, BP5, BP6: real;
                ii: integer;
                BPOIS: array[1..mnp, 1..6] of real;
{**********************************************************************}

{~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~}
{~}
{    Remember to reset the value of m, the number of fitting variables.    }
{ }
{~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~}
{~}

        function f (x: vector; d: datarow): real; {x(1..m) the parameters,}
{                   d has the data}
                var
                        k, j: integer;
        begin
                k := 0;
                for j := 1 to np do
                        begin
                                if abs(BPOIS[j, 1] - d[1]) <= 0.1 then
                                        k := j;
```

```
                    end;
            if k = 0 then
                    writeln(dout, 'function error');
            f := x[1] * BPOIS[k, 2] + x[2] * BPOIS[k, 3] + x[3] * BPOIS[k, 4] + x[4] * BPOIS[k, 5] + x[5] * BPOIS[k, 6];
    end;

{------------------------------------------------------------------------------}
{-}

    procedure initialize;           {initialize the value of simp to 0.}
{0}

            var
                    i, j: index;
    begin
            for i := 1 to n do
                    begin
                            for j := 1 to n do
                                    simp[i, j] := 0.0;
                    end  {i loop}
    end;    {initialize}

{------------------------------------------------------------------------------}
{-}

    procedure sum_of_residuals (var x: vector);
  {computes the sum of the squares of the residuals}
  {x(1..m) passes parameters. Result returned in x(n)}
            var
                    i: index;
    begin
            x[n] := 0.0;
            for i := 1 to np do
                    begin
                            x[n] := x[n] + sqr(f(x, data[i]) - data[i, 2]);
      { x[n] := x[n] + sqr( (f(x,data[i]) - data[i,2])/f(x,data[i]));}
                    end;  {loop}
    end;  {sum_of_residuals}

{------------------------------------------------------------------------------}
{-}

    procedure enter;            {enters data from disk file fname. file}
{           must terminate with EOF immediately after}
{           last number. data in the order:}
{            -maximum number iterations,}
{            -starting point coordinates}
{            -starting increments}
{           -data}

            var
                    i, j: index;
                    temp: real;
    begin   {enter}
            write(dout, ' SIMPLEX optimization version ');
            write(dout, date);
            writeln(dout, ' @ mc/bc fsu');
            writeln(dout, memo);
            writeln(dout, ' accessing file GOAL.DAT2');
            writeln(dout);
            read(din, maxiter);
            writeln(dout, ' max number of itertions is := ', maxiter : 5);
            write(dout, ' start coord.: ');
            for i := 1 to m do
                    begin
                            if eoln(din) then
                                    readln(din);
                            read(din, temp);
                            simp[1, i] := temp;
                            if (i mod lw) = 0 then
                                    writeln(dout);
                            write(dout, simp[1, i])
```

```
            end;
        writeln(dout);
        write(dout, ' start steps: ');
        for i := 1 to m do
            begin
                if eoln(din) then
                    readln(din);
                read(din, temp);
                step[i] := temp;
                if (i mod lw) = 0 then
                    writeln(dout);
                write(dout, step[i])
            end;
        writeln(dout);
        write(dout, ' max. errors: ');
        for i := 1 to n do
            begin
                if eoln(din) then
                    readln(din);
                read(din, temp);
                maxerr[i] := temp;
                if (i mod lw) = 0 then
                    writeln(dout);
                write(dout, maxerr[i])
            end;
        writeln(dout);
        writeln(dout, ' data: ');
        writeln(dout, ' x' : 14, ' y' : 14);
        np := 0;
        while not eof(din) do
            begin
                np := succ(np);
                write(dout, ' #', np : 3);
                for j := 1 to nvpp do
                    begin
                        read(din, temp);
                        if eoln(din) then
                            readln(din);
                        data[np, j] := temp;
                        write(dout, data[np, j] : 15)
                    end;
                writeln(dout);
            end;      {while}
        np := np - 1      {my fix to make the i/o work on the Lisa}
    end;        {enter}

{------------------------------------------------------------------------}
{}

    procedure report;
        var
            y, dy, sigma: real;
            d1, d2: text;       {disk out files}
            i, j: index;
    begin
        writeln(dout, ' program exited after', niter : 5, ' iterations ');
        writeln(dout, ' the final simplex is');
        for j := 1 to n do
            begin
                for i := 1 to n do
                    begin
                        if (i mod lw) = 0 then
                            writeln(dout);
                        write(dout, simp[j, i] : 10);
                    end;
                writeln(dout);
            end;        {do j}
        writeln(dout, ' the mean is');
        for i := 1 to n do
            begin
                if (i mod lw) = 0 then
                    writeln(dout);
```

```
                        write(dout, mean[i]);
                end;
        writeln(dout);
        writeln(dout, ' the estimated fractional error is');
        for i := 1 to n do
                begin
                        if (i mod lw) = 0 then
                                writeln(dout);
                        write(dout, error[i]);
                end;
        writeln(dout);
        writeln(dout, ' #' : 4, 'x' : 10, 'y' : 15, 'y"' : 15, 'dy(%)' : 15);
        sigma := 0.0;
        for i := 1 to np do
                begin
                        y := f(mean, data[i]);
                        dy := data[i, 2] - y;
                        if y <> 0 then
                                dy := 100 * dy / y
                        else if data[i, 2] <> 0 then
                                dy := 100;
                        sigma := sigma + sqr(dy);
                        writeln(dout, i : 4, data[i, 1] : 15, data[i, 2] : 15, y : 15, dy : 15);
                end;
        sigma := sqrt(sigma / np);
        writeln(dout, ' the standard deviation is', sigma);
        sigma := sigma / sqrt(np - m);
        write(dout, ' the estimated error of the');
        writeln(dout, ' function is', sigma);
    end;    {report}

{-----------------------------------------------------------------------}
{~}

    procedure first;
        var
            i, j: index;
    begin
        writeln(dout, ' starting simplex');
        for j := 1 to n do          {vertices}
                begin
                        write(dout, ' simp[', j : 1, ']');
                        for i := 1 to n do
                                begin           {dimensions}
                                        if (i mod lw) = 0 then
                                                writeln(dout);
                                        write(dout, simp[j, i])
                                end;            {dimensions}
                        writeln(dout)
                end;            {vertices}
        writeln(dout)
    end;            {first}

{-----------------------------------------------------------------------}
{~}

    procedure new_vertex;           {next in place of the worst vertex}
        var
            i, j: index;
    begin
        write(dout, ' -- ', niter : 4);
        for i := 1 to n do
                begin
                        simp[h[n], i] := next[i];
                        write(dout, next[i])
                end;
        writeln(dout)
    end;            {new_vertex}

{-----------------------------------------------------------------------}
{'}
```

```
procedure order;                {gives high/low in each parameter}
        {in simp. caution: not initialized}
        var
                i, j: index;
begin
        for j := 1 to n do        {all dimensions}
                begin
                        for i := 1 to n do        {of all vertices}
                                begin            {find best and worst}
                                        if simp[i, j] < simp[l[j], j] then
                                                l[j] := i;
                                        if simp[i, j] > simp[h[j], j] then
                                                h[j] := i
                                end;            {i loop}
                end;            {j loop}
end;            {order}

{---------------------------------------------------------------------------}
{~}

begin            {simplex}
        initialize;            {initialize the simplex matrix}
        writeln;
   {reset(din,fname);}
        {fname is on disk}
  {rewrite(dout,'-console');}
     {output goes to console}
  {enter;}
            {get the data}
        OPEN(din, 'goal.dat2');        {reset in file}
        OPEN(din1, 'composite Bz2');
        OPEN(dout, 'SIMPLEX.FIT2');    {output file}
        RESET(DIN);
        RESET(DIN1);
        REWRITE(DOUT);

        enter;
        for ii := 1 to np do
                begin
                        readln(din1, BP1, BP2, BP3, BP4, BP5, BP6);
                        BPOIS[ii, 1] := BP1;
                        BPOIS[ii, 2] := BP2;
                        BPOIS[ii, 3] := BP3;
                        BPOIS[ii, 4] := BP4;
                        BPOIS[ii, 5] := BP5;
                        BPOIS[ii, 6] := BP6;
                end;

        sum_of_residuals(simp[1]);        {first vertex}

        for i := 1 to m do            {compute offset of the vertices}
                begin                {of the starting simplex}
                        p[i] := step[i] * (sqrt(n) + m - 1) / (m * root2);
                        q[i] := step[i] * (sqrt(n) - 1) / (m * root2)
                end;

        for i := 2 to n do            {all vertices of the starting simplex}
                begin
                        for j := 1 to m do
                                simp[i, j] := simp[1, j] + q[j];
                        simp[i, i - 1] := simp[1, i - 1] + p[i - 1];
                        sum_of_residuals(simp[i])        {and their residuals}
                end;

        for i := 1 to n do            {preset}
                begin
                        l[i] := 1;
                        h[i] := 1
                end;                {before calling}
        order;
```

```
        first;                  {pass to printer}
  {rewrite(dout,'-console');}
      {and}
  {first;}
           { to the screen}

        niter := 0;                  { no iterations yet}

        repeat                       {keep iterating}
                done := true;
                niter := succ(niter);

                for i := 1 to n do
                        center[i] := 0.0;
                for i := 1 to n do           {compute centroid}
                        if i <> h[n] then              {excluding the worst}
                                for j := 1 to m do
                                        center[j] := center[j] + simp[i, j];

                for i := 1 to n do           {first attemp to reflect}
                        begin
                                center[i] := center[i] / m;
                                next[i] := (1.0 + alfa) * center[i] - alfa * simp[h[n], i]
   {next vertex is the specular reflection of the worst}
                        end;
                sum_of_residuals(next);

                if next[n] <= simp[l[n], n] then
                        begin                {better than the best ?}
                                new_vertex;              {accepted}
                                for i := 1 to m do         {and expanded}
                                        next[i] := gamma * simp[h[n], i] + (1.0 - gamma) * center[i];
                                sum_of_residuals(next);        {still better ?}
                                if next[n] <= simp[l[n], n] then
                                        new_vertex
                        end               {expansion accepted}

                else                   {if not better than the best}
                        begin
                                if next[n] <= simp[h[n], n] then
                                        new_vertex                  {better than worst}
                                else                 {worse than worst}
                                        begin               {then: contract}
                                                for i := 1 to m do
                                                        next[i] := beta * simp[h[n], i] + (1.0 - beta) *
center[i];
                                                sum_of_residuals(next);
                                                if next[n] <= simp[h[n], n] then
                                                        new_vertex              {contraction accepted}
                                                else              {if still bad}
                                                        begin              {shrink all bad vertices}
                                                                for i := 1 to n do
                                                                        begin
                                                                                for j := 1 to m do
                                                                                        simp[i, j]
:= (simp[i, j] + simp[l[n], j]) * beta;

        sum_of_residuals(simp[i])
                                                                        end           {i loop}
                                                        end              {else}
                                        end              {else}
                        end;              {else}

                order;
                for j := 1 to n do           {check for convergence}
                        begin
                                error[j] := (simp[h[j], j] - simp[l[j], j]) / simp[h[j], j];
                                if done then
                                        if error[j] > maxerr[j] then
                                                done := false
                        end
```

```
            until (done or (niter = maxiter));

            for i := 1 to n do              {average each parameter}
                    begin
                            mean[i] := 0.0;
                            for j := 1 to n do
                                    mean[i] := mean[i] + simp[j, i];
                            mean[i] := mean[i] / n
                    end;

    {report;}
              {to console}
    {rewrite(dout,'-printer');}
        {and do it again}
            report;                  {to the printer}
            writeln(dout, chr(page));
            close(dout);
end.                {of simplex}
```

```
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c                                                                    c
c                                                                    c
c                Load Macintosh ToolBox trap file                    c
c                                                                    c
!!M InLines.f
c                                                                    c
c                                                                    c
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c                                                                    c
c                                                                    c
                          PROGRAM INVHLX
c                                                                    c
c                                                                    c
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
c                                                                    c
c                                                                    c
c                                                                    c
c                      ***********************                       c
c                      *                     *                       c
c                      *   PROGRAM SYNOPSIS  *                       c
c                      *                     *                       c
c                      ***********************                       c
c                                                                    c
c     Program INVHLX ( inverse helix ): This program assumes an axial c
c     profile for the entrance magnetic field of a bifilar helix and  c
c     attempts to derive a radial profile that will produce the       c
c     desired field taper. Infinite length helix formulas are used    c
c     to estimate the on axis field. The output provides information  c
c     for a Biot-Savart calculation. Calculations are based on:       c
c                                                                    c
c                r(z) = ru*[K1'(ru)]*f(z)/[K1'(r(z))]                 c
c                                                                    c
c     A user supplied field profile is used to determine the helix    c
c     radial profile. The infinite helix formula for on-axis field    c
c     strength vs. coil radius is assumed. Starting at the beginning  c
c     of the uniform radius helix section and stepping backwards      c
c     a Newton-Raphson method is used to invert the dK1(x)/dx modified c
c     Bessel function for the required value of x. The calculation    c
c     can not be carried all the way back, but can be carried to      c
c     within a few steps of the zero point.                           c
c                                                                    c
c     Matching at the end of the helix taper is achieved by assuming  c
c     a constant angle cone which extends the taper back to z=0 with  c
c     the same slope as the last inverted point which was calculated. c
c                                                                    c
c     Both R(z) and dR(z)/dz are output for use in a Biot-Savart code c
c     in order to calculate the true field profile on and off axis.   c
c                                                                    c
C====================================================================C
c                                                                    c
c                                                                    c
c                                                                    c
c                      ************************                      c
c                      *                      *                      c
c                      *   NUMERICAL METHODS  *                      c
c                      *                      *                      c
c                      ************************                      c
c                                                                    c
c     MODIFIED BESSEL FUNCTION K:                                     c
c     The on-axis helix field strength depends on the derivitive of   c
c     the modified Bessel function of the second kind, K1, with       c
c     respect to its argument. The Bessel functions are calculated    c
c     using the polynomial approximations found in Abramowitz and     c
c     Stegun, "Handbook of Mathematical Functions", Dover 9/e, 1972   c
c                                                                    c
c                                                                    c
c     NEWTON-RAPHSON ROOT FINDER:                                     c
c     The Bessel function is inverted to find the helix radius which  c
c     will give the desired field strength. The inversion is performed c
c     by a Newton-Raphson root finding calculation which determines   c
c     the radius necessary for the given field value. Set             c
```

```
C                                                                      C
C                    F(x) = x[K1'(x)] - f(z)*xu[K1'(xu)]               C
C     which gives:                                                     C
C                          x = x - [F/F']                              C
C                                                                      C
C======================================================================C
C                                                                      C
C                                                                      C
C                                                                      C
C                          ************************                    C
C                          *                      *                    C
C                          *     !!! NOTES !!!    *                    C
C                          *                      *                    C
C                          ************************                    C
C                                                                      C
C                                                                      C
C     1.     Unless otherwise stated, all values are in either         C
C            normalized or cgs units.                                  C
C                                                                      C
C======================================================================C
C                                                                      C
C                                                                      C
C                                                                      C
C                          ************************                    C
C                          *                      *                    C
C                          * SUBROUTINES/FUNCTIONS *                   C
C                          *                      *                    C
C                          ************************                    C
C                                                                      C
C                                                                      C
C     KBES(X,N,+/- 1) - Function which calculates K modified Bessel    C
C                       function. X is the argument, N is the order    C
C                       of K, +1 specifies that Kn(x) is to be         C
C                       calculated, and -1 specifies that dKn(x)/dx    C
C                       is to be returned. Calculations are based on   C
C                       the polynomial approximations given in         C
C                       Abramowitz and Stegun, "Handbook of            C
C                       Mathematical Functions", Dover 9/e, 1972.      C
C                                                                      C
C     FTAPER(Z,       - Function which calculates the helix field      C
C            ZTAPER,    profile at axial position z ( note that z is   C
C            FOPT)      is in normalized units of z(cm)/LHELIX(cm) .)  C
C                       ZTAPER is the number of helix periods the      C
C                       field builds up over. FOPT controls the        C
C                       built in taper profile options in FTAPER.      C
C                       Built in options are:                          C
C                       FOPT     TAPFUN     TYPE                       C
C                        1       'CUBI'     CUBIC FIT,ZERO END d/dz    C
C                        2       'COS'      (1-COS)/2                  C
C                        3       'POLY'     POLYNOMIAL,SUPPLY COEFS    C
C                        4       'USER'     USER SUPPLIES              C
C                                           ENTIRE FTAPER FUNCTION     C
C                                                                      C
C======================================================================C
C                                                                      C
C                                                                      C
C                                                                      C
C                          ************************                    C
C                          *                      *                    C
C                          * VARIABLE DEFINITIONS *                    C
C                          *                      *                    C
C                          ************************                    C
C                                                                      C
C                                                                      C
C     VARIABLE                 DEFINITION                              C
C     --------                 ----------                              C
C     ALPHA
C     AREA
C     AZ
C     AZM
C     AZP
C     B
```

```
C      BFIELD1
C      BFIELD2
C      B_PERP
C      CFUN
C      CN
C      CNFLG
C      CONE
C      COUT
C      CUR
C      CURRENT1
C      CURRENT2
C      D1
C      D2
C      DELTA_R
C      DETA
C      DFDX
C      DFMDX
C      DFPDX
C      DR
C      DRDZ
C      DXDZ
C      DZ
C      ERR
C      ERROR
C      ETA
C      ETA1
C      F
C      FM
C      FOPT
C      FP
C      F_THETA
C      FZ
C      F0
C      GO
C      I,J,K
C      ICONE
C      IEND
C      IN
C      INTCPT
C      INTERACTIVE
C      INTFLG
C      IPTS
C      I_SUPPLY
C      JPTS
C      JSTART
C      JSTOP
C      KHELIX
C      KINTCP
C      KTEST
C      K0
C      K0M
C      K0P
C      KW2
C      K1
C      K1M
C      K1P
C      LHELIX
C      LISA
C      L1_LAYER
C      L1_WIRE
C      L2_LAYER
C      L2_WIRE
C      L_FRACT
C      L_LOOP
C      L_TAPER
C      L_UNIFORM
C      M
C      MERR
C      MRATIO
C      MXCOEF
C      MXERR
C      MXINCP
```

```
C     MXITER
C     MXLAYERS
C     MXPTS
C     N
C     NCOEF
C     NITER
C     NLISA
C     NPTS
C     NWIRES
C     N_UNIFORM
C     OUT
C     PERR
C     PI
C     PRATIO
C     P1KW
C     P2KW
C     R
C     RARC
C     RATIO
C     RC
C     RCONE
C     RESIST
C     RHELIX
C     RHO
C     RINTCP
C     RTEST
C     RU
C     R1_TOTAL
C     R2_TOTAL
C     R_PRIME
C     S
C     SLOPE
C     SUM
C     SUM1
C     SUM2
C     SUM3
C     SUM_CETA
C     SUPPLY
C     TAPFUN
C     TEMP
C     TWPI
C     T_INSULATION
C     VOLT1
C     VOLT2
C     V1
C     V2
C     V_SUPPLY
C     X
C     XM
C     XMNEW
C     XNEW
C     XP
C     XPNEW
C     XU
C     X_KPRIME
C     Z
C     ZARC
C     ZC
C     ZCONE
C     ZDIF
C     ZHELIX
C     ZM
C     ZO
C     ZP
C     ZSTOP
C     ZTAPER
C                                                                      C
C======================================================================C
C                                                                      C
C                                                                      C
C                                                                      C
C                    ********************************                  C
```

```
C                    *                               *                 C
C                    * PARAMETERS,TYPES,DIMENSIONS *                   C
C                    *                               *                 C
C                    *******************************                   C
C                                                                      C
C                                                                      C
      PARAMETER ( MXITER=30, MXINCP=5, MXCOEF=10, MXPTS=1000 )
      PARAMETER ( MXLAYERS=20 )
C
      INTEGER      CONE,FOPT,OUT,DIAGNO,ICONE
      INTEGER      TICK_START
C
      REAL         LHELIX,MXERR,KHELIX,K0,K1,K0M,K1M,K0P,K1P
      REAL         KBES,MRATIO,MERR,INTCPT(MXINCP)
      REAL         L_FRACT,N_UNIFORM,I_SUPPLY,KW2
      REAL         L_TAPER,L_UNIFORM,L_LOOP,L1_WIRE,L2_WIRE
      REAL         L1_LAYER,L2_LAYER,L1_TOTAL,L2_TOTAL
      REAL         MIL_TO_CM
C
      REAL*4       RUN_START,RUN_END,TICK_END
C
      CHARACTER*4  TAPFUN,GO
      CHARACTER*1  TAB,KEY
      CHARACTER*8  RUN_TIME
      CHARACTER*9  RUN_DATE
      CHARACTER*32 IN_FILE
C
      LOGICAL      CNFLG,INTFLG,DFLG,INTERACTIVE,SUPPLY
C
      DIMENSION    RINTCP(MXINCP),CFUN(MXCOEF),RHELIX(MXPTS)
      DIMENSION    DRDZ(MXPTS),ZHELIX(MXPTS),RCONE(MXINCP)
      DIMENSION    ZCONE(MXINCP),SLOPE(MXINCP),ICONE(MXINCP)
      DIMENSION    ZARC(MXPTS),RARC(MXPTS),R_PRIME(MXPTS)
      DIMENSION    L1_TOTAL(MXLAYERS),L2_TOTAL(MXLAYERS)
      DIMENSION    CN(MXLAYERS),L1_WIRE(MXLAYERS)
      DIMENSION    L2_WIRE(MXLAYERS),L1_LAYER(MXLAYERS)
      DIMENSION    L2_LAYER(MXLAYERS),R1_TOTAL(MXLAYERS)
      DIMENSION    R2_TOTAL(MXLAYERS),VOLT1(MXLAYERS)
      DIMENSION    VOLT2(MXLAYERS),CURRENT1(MXLAYERS)
      DIMENSION    CURRENT2(MXLAYERS),BFIELD1(MXLAYERS)
      DIMENSION    BFIELD2(MXLAYERS),P1KW(MXLAYERS),P2KW(MXLAYERS)
C                                                                      C
C======================================================================C
C                                                                      C
C                                                                      C
C                                                                      C
C                       *************************                      C
C                       *                       *                      C
C                       *        COMMONS        *                      C
C                       *                       *                      C
C                       *************************                      C
C                                                                      C
C                                                                      C
      COMMON/FUNCTN/ NCOEF,CFUN
C                                                                      C
C======================================================================C
C                                                                      C
C                                                                      C
C                                                                      C
C                       *************************                      C
C                       *                       *                      C
C                       *        FORMATS        *                      C
C                       *                       *                      C
C                       *************************                      C
C                                                                      C
C                                                                      C
    1 FORMAT('1')
    2 FORMAT('0',9X,'INVERSE HELIX PROFILE PROGRAM: VERSION MARCH 1985')
    3 FORMAT('0',' ******************** ECHO CHECK OF INPUT DATA '
     3       ,'********************')
    4 FORMAT('0',' PRINT VALUES FOR LISA GRAPHICS EVERY ',I2
     4       ,' POINTS, STORE IN FILE FOR',I2,/1X,' DATA IS READ FROM'
```

```
    4       ,' FILE FOR',I2,' AND WRITTEN TO FILE FOR',I2,/1X
    4       ,'  DIAGNOSTIC PRINTS REQUESTED?',L4,', IF SO PRINT TO'
    4       ,' FILE FOR',I2,/1X,'  DATA FOR CONE EXTENSIONS TO THE'
    4       ,' END OF THE HELIX START IN FILE FOR',I2,' AND SUCCESSIVE'
    4       ,' CONE TAPERS CAN BE FOUND IN SUCCESSIVE FILES.')
  5 FORMAT('0','  HELIX PERIOD (CM) = ',F10.4,//1X,'  HELIX RADIUS'
    5       ,' IN UNIFORM SECTION (CM) = ',F10.5,//1X,'  HELIX PERIODS'
    5       ,' OVER WHICH THE RADIUS IS TAPERED (Z/L) = ',F10.4)
  6 FORMAT('0','  TYPE OF HELIX FIELD TAPER : ',A4)
  7 FORMAT('0','  NUMBER OF TERMS IN THE POLYNOMIAL =',I3
    7       ,(/1X,'  CFUN(',I2,') = ',E14.7))
  8 FORMAT('0','  FRACTION OF HELIX PERIOD FOR BACKWARD STEPPING = '
    8       ,F10.6,//1X,'  STOP STEPPING WHEN Z = ',F10.5)
  9 FORMAT('0','  RADIAL VALUES (CM) AT THE START OF THE HELIX FOR'
    9       ,' CONNICAL EXTENSION TO END TAPER : ',/1X
    9       ,(/1X,'  INTERCEPT(',I2,') = ',F10.5))
 10 FORMAT('0','  NEWTON-RAPHSON CONTROLS :',/1X
    1       ,'  MAXIMUM NUMBER OF ITERATIONS = ',I3,/1X
    1       ,'  MAXIMUM ALLOWED RELATIVE ERROR = ',E12.5)
 11 FORMAT('0',' ******************** END OF ECHO CHECK, START'
    1       ,' CALCULATIONS ********************')
 12 FORMAT('0','  RADIAL INTERCEPT VALUES AFTER SORTING :',/1X
    2       ,(/1X,'  INTERCEPT(',I2,') = ',F10.5))
 13 FORMAT('0','  TAPER PROFILE OPTION NOT ALLOWED; HALT!')
 14 FORMAT('0','  FOPT = ',I2)
 15 FORMAT('0',' REQUESTED NUMBER OF POINTS EXCEEDS MAXIMUM: '
    5       ,'N = ',I5,/2X,'CHANGE REQUEST OR MXPTS IN PARAMETER'
    5       ,' STATEMENT.')
 16 FORMAT('0','  NPTS = ',I5,' X = ',F10.5,'  XU*K1''(XU) = ',E14.7)
 17 FORMAT('0','  Z = ',F10.5,'  F(Z) = ',E12.5,' XU*K1"(XU)*F(Z) = '
    7       ,E12.5)
 18 FORMAT('0','J','X','K0(X)','K1(X)','F(X)'
    8       ,'DFDX','F/DFDX','ERROR')
 19 FORMAT(' ',2X,I5,7E14.5)
 20 FORMAT('0',5X,'BESSEL FUNCTION INVERSION HAS FAILED!',/' ',5X
    2       ,'Z = ',E12.5,5X,'TWO-PI*R/L = ',E12.5,5X,'ERROR = '
    2       ,E12.5)
 21 FORMAT('0',1X,I5,1X,'R = ',E12.5,1X,'Z = ',E12.5,1X,'SLOPE = '
    1       ,E12.5,1X,'INTERCEPT = ',E12.5)
 22 FORMAT(' ',2X,I5,2X,F10.5,2X,F10.5,2X,E12.5,2X,I5)
 23 FORMAT(' ',2X,F10.5,2X,F10.5,2X,E12.5)
 24 FORMAT('0', '  THE REQUESTED NUMBER OF POINTS',I5
    4       ,' EXCEEDS THE ALLOWED MAXIMUM ',I5)
 25 FORMAT('0',' SELECT HELIX TAPER, ALLOWED VALUES ARE 0 THRU'
    5       ,I2,' : ')
 26 FORMAT('0','  TAPER VALUE',I2,' IS OUT OR RANGE. SELECT VALUE'
    6       ,' WITHIN THE ALLOWED RANGE.')
 27 FORMAT('0','INPUT WIRE TEMPERATURE, BULK RESISTIVITY, AND'
    7       ,' RESISTANCE TEMPERATURE COEF :')
 28 FORMAT('0','INPUT PERIODS IN UNIFORM HELIX, AND FRACTION OF'
    8       ,' PERIOD OCCUPIED BY COILS :')
 29 FORMAT('0','INPUT DESIRED B FIELD (GAUSS), SUPPLY CURRENT LIMIT'
    9       ,' AND VOLTAGE LIMIT :')
 30 FORMAT('0','INPUT WIRE WIDTH (DIA), HEIGHT, AND INSULATION'
    &       ,' THICKNESS (ALL IN MILS) :')
 31 FORMAT('0','  # OF PERIODS IN TAPER = ',F5.2,/'   # OF PERIODS'
    1       ,' IN UNIFORM SECTION = ',F5.2,/'   FRACTION OF PERIOD'
    1       ,' USED FOR COIL WINDING = ',F6.4)
 32 FORMAT('0','  # OF WIRES IN ONE LAYER = ',I4,/'   WIRE'
    2       ,' WIDTH (CM) = ',F7.4,/'   WIRE HEIGTH (CM) = ',F7.4
    2       ,/'   INSULATION THICKNESS (CM) = ',F7.5)
 33 FORMAT('0','  BULK RESISTIVITY = ',E12.4,'  TEMP COEF = ',E12.4
    3       ,/'   ASSUMED WIRE TEMP (DEG C) = ',F6.1)
 34 FORMAT('0','  THE DESIRED PERPENDICULAR MAGNETIC FIELD (GAUSS) ='
    4       ,F8.1)
 35 FORMAT('0','  POWER SUPPLY LIMITS: I (AMPS) = ',F7.1
    5       ,'  V (VOLTS) = ',F7.1)
 36 FORMAT('0',' VALUES WITH ONLY ONE TAPER INCLUDED:',/'  LAYERS'
    6       ,' G/AMP    AMPS    B(GAUSS    L (CM)    R(OHMS)    '
    6       ,'VOLTS     POWER(KW)')
 37 FORMAT('0',' VALUES WITH BOTH TAPERS INCLUDED:',/'  LAYERS'
    7       ,' G/AMP    AMPS    B(GAUSS    L (CM)    R(OHMS)    '
```

```
     7        ,'VOLTS    POWER(KW)')
  38 FORMAT(' ',2X,I2,3X,F6.3,1X,F7.1,2X,F7.1,2X,F8.1,2X,F7.4,2X,F7.1
     8        ,3X,F8.1)
  39 FORMAT('  >>>>>>>>>>>>>>>>>>>>> INVERSE HELIX'
     &       ,' PROFILE CODE <<<<<<<<<<<<<<<<<<<<<'
     &       ,// '     |          Run Date - ',A9
     &       ,' | Run Time - ',A8,'       |',/)
  40 FORMAT(' ',' TIME FOR CALCULATIONS (SECONDS)  = ',EN12.3,/
     &            ,'  TIME FOR CALCULATIONS (Ticks/60) = ',EN12.3)
C                                                                      C
C======================================================================C
C
C
C     FIND THE DATE AND TIME OF THE SESSION
C
      CALL DATE( RUN_DATE )
      CALL TIME( RUN_TIME )
C                                                                      C
C======================================================================C
C                                                                      C
C                                                                      C
C                                                                      C
C                        *************************                     C
C                        *                       *                     C
C                        *     INPUT/OUTPUT      *                     C
C                        *                       *                     C
C                        *************************                     C
C                                                                      C
C                                                                      C
      IN     = 10
      OUT    = 11
      LISA   = 15
      CONE   = 20
      DIAGNO =  7
C
C     OPEN FILE FOR I/O
C
      OPEN( UNIT = 10, STATUS = 'OLD', FILE =      'FOR010.DAT' )
      OPEN( UNIT = 11, STATUS = 'NEW', FILE =      'INVHLX.OUT' )
      OPEN( UNIT = 15, STATUS = 'NEW', FILE =       'GRAPH.OUT' )
      OPEN( UNIT = 20, STATUS = 'NEW', FILE =        'CONE.OUT' )
      OPEN( UNIT = 21, STATUS = 'NEW', FILE =       'CONE1.OUT' )
      OPEN( UNIT =  7, STATUS = 'NEW', FILE = 'DIAGNOSTICS.OUT' )
C
      WRITE(OUT, 1)
      WRITE(OUT, 2)
C                                                                      C
C======================================================================C
C                                                                      C
C                                                                      C
C                                                                      C
C                        *************************                     C
C                        *                       *                     C
C                        *   CONSTANTS/DEFAULTS  *                     C
C                        *                       *                     C
C                        *************************                     C
C                                                                      C
      PI          = 3.1415926535
      TWPI        = 2.0*PI
      PID2        = PI/2.0
      MIL_TO_CM   = 0.001*2.54
C
      NCOEF       =  1
      NLISA       =  5
      LHELIX      =  2.54
      RU          =  0.95
      ZTAPER      =  5.0
      ZSTOP       =  0.1
      NITER       =  10
      MXERR       =  1.0E-5
      TAPFUN      = 'CUBI'
      DZ          =  0.01
```

```
      FOPT        =  1
      KINTCP      =  0
      KTEST       =  0
      CNFLG       = .FALSE.
      DFLG        = .FALSE.
      INTFLG      = .FALSE.
      INTERACTIVE = .FALSE.
      GO          = 'N'
C
      DO K=1,MXCOEF
      CFUN(K) = 0.0
      END DO
C
      DO K=1,MXINCP
      RINTCP(K) = 0.0
      END DO
      RINTCP(1) = 3.0
C                                                                      C
C                                                                      C
C----------------------------------------------------------------------C
C                                                                      C
C >>>>>>>>>>>>>>>>>>>>>> BEGIN PROGRAM INVHLX <<<<<<<<<<<<<<<<<<<<<<< C
C                                                                      C
C----------------------------------------------------------------------C
C
C
C     SET INITIAL TIME FOR START OF CALCULATIONS, IN SECONDS
C
      RUN_START  = SECNDS( 0.0 )
      TICK_START = TickCount
C
C
C     READ IN I/O AND DIAGNOSTIC CONTROLS
C
      READ(IN, *) INTERACTIVE
      READ(IN, *) NLISA,DFLG,IN,OUT,DIAGNO,LISA,CONE
C
C     READ IN HELIX DESIGN VALUES
C
      READ(IN, *) LHELIX,RU,RINTCP
      READ(IN, *) ZTAPER,DZ,ZSTOP
C
C     READ IN CONTROLS FOR NEWTON-RAPHSON INVERSION OF K1(X)
C
      READ(IN, *) NITER,MXERR
C
C     READ IN TYPE OF TAPER PROFILE, AND COEFFICIENTS IF ANY
C
      READ(IN, *) TAPFUN,NCOEF,CFUN
C
C     ECHO CHECK INPUT VALUES
C
      WRITE(OUT, 3)
      WRITE(OUT, 4) NLISA,LISA,IN,OUT,DFLG,DIAGNO,CONE
      WRITE(OUT, 5) LHELIX,RU,ZTAPER
      WRITE(OUT, 6) TAPFUN
      IF ( TAPFUN .EQ. 'POLY' ) THEN
      WRITE(OUT, 7) NCOEF,(I,CFUN(I),I=1,NCOEF)
      END IF
      WRITE(OUT, 8) DZ,ZSTOP
      WRITE(OUT, 9) (I,RINTCP(I),I=1,MXINCP)
      WRITE(OUT,10) NITER,MXERR
      WRITE(OUT,11)
C
C     END ECHO CHECK PRINT
C
C     SET HELIX PARAMETERS FOR INTERNAL CODE USE
C
      DO I=1,MXINCP
      IF ( RINTCP(I) .NE. 0.0 ) THEN
      KINTCP = KINTCP + 1
      CNFLG  = .TRUE.
```

```
      KTEST  = 1
      END IF
      END DO
C
      IF ( CNFLG ) THEN
      INTFLG = .TRUE.
      IEND   = KINTCP - 1
C
      DO I=1,IEND
C
      RTEST  = RINTCP(I)
      JSTART = I + 1
C
      DO J=JSTART,KINTCP
      IF ( RINTCP(J) .LT. RTEST ) THEN
      RTEST      = RINTCP(I)
      RINTCP(I) = RINTCP(J)
      RINTCP(J) = RTEST
      RTEST      = RINTCP(I)
      END IF
      END DO
C
      END DO
      END IF
C
      WRITE(OUT,12) (I,RINTCP(I),I=1,MXINCP )
C
      IF ( TAPFUN .EQ. 'CUBI' ) FOPT = 1
C
      IF ( TAPFUN .EQ. 'COS'  ) FOPT = 2
C
      IF ( TAPFUN .EQ. 'POLY' ) FOPT = 3
C
      IF ( TAPFUN .EQ. 'USER'  ) FOPT = 4
C
      IF ( ( FOPT .LE. 0 ) .OR. ( FOPT .GE. 5 ) ) THEN
      WRITE(OUT,13)
      GO TO 200
      END IF
C
      WRITE(OUT,14) FOPT
C
      KHELIX = TWPI/LHELIX
      XU     = KHELIX*RU
C
C     INITIALIZE BACKSTEP LOOP PARAMETERS
C
      X      = XU
      XP     = XU
      XM     = XU
      F0     = XU*KBES(XU,1,-1)
      FZ     = 1.0
C
C     CALCULATE THE NUMBER OF BACKWARD STEPS
C
      NPTS   = INT( ( ZTAPER - ZSTOP )/DZ )
      IF ( NPTS .GT. MXPTS ) THEN
      WRITE(OUT,15) NPTS
      GO TO 200
      END IF
C
      WRITE(OUT,16) NPTS,X,F0
C
C     ENTER LOOP OVER THE NUMBER OF AXIAL POINTS IN TAPER
C
C     !!!!! BEGIN POINTS !!!!!
C
      DO I=1,NPTS
C
      Z      = ZTAPER - FLOAT( I )*DZ
      ZP     = Z + DZ/10.0
      ZM     = Z - DZ/10.0
```

```
C
      FZ      = FTAPER( Z,ZTAPER,FOPT  )
      FP      = FTAPER( ZP,ZTAPER,FOPT )
      FM      = FTAPER( ZM,ZTAPER,FOPT )
C
C     SET VARIABLES FOR BESSEL FUNCTION INVERSION
C
      AZ      = F0*FZ
      AZP     = F0*FP
      AZM     = F0*FM
C
C     DIAGNOSTIC PRINT
C
      IF ( DFLG ) THEN
      WRITE(DIAGNO,17) Z,FZ,AZ
      WRITE(DIAGNO,18)
      END IF
C
C     INVERT K1' BESSEL FUNCTION
C
C     !!!!! BEGIN INVERSION !!!!!
C
      Dr ᴊ=1,NITER
C
      K0      = KBES(X,0,1)
      K1      = KBES(X,1,1)
C
      K0M     = KBES(XM,0,1)
      K1M     = KBES(XM,1,1)
C
      K0P     = KBES(XP,0,1)
      K1P     = KBES(XP,1,1)
C
      F       = X*K0 + K1 + AZ
      DFDX    = (X + 1.0/X)*K1
C
      FM      = XM*K0M + K1M + AZM
      DFMDX   = (XM + 1.0/XM)*K1M
C
      FP      = XP*K0P + K1P + AZP
      DFPDX   = (XP +1.0/XP)*K1P
C
      RATIO   = F/DFDX
      MRATIO  = FM/DFMDX
      PRATIO  = FP/DFPDX
C
C     NOTE THAT USE HAS BEEN MADE OF THE NEGATIVE VALUE
C     OF THE DERIVITIVE OF THE BESSEL FUNCTION SO THAT
C     X - F/F' CHANGES TO X + ABS(F)/F'
C
      XNEW    = X  + RATIO
      XMNEW   = XM + MRATIO
      XPNEW   = XP + PRATIO
C
      ERR     = 0.5*ABS( ( XNEW - X )/( XNEW + X ) )
      MERR    = 0.5*ABS( ( XMNEW - XM )/( XMNEW + XM ) )
      PΞRR    = 0.5*ABS( ( XPNEW - XP )/( XPNEW + XP ) )
C
      X       = XNEW
      XM      = XMNEW
      XP      = XPNEW
C
      ERROR   = ERR
      IF ( MERR .GT. ERROR ) ERROR = MERR
      IF ( PERR .GT. ERROR ) ERROR = PERR
C
C     DIAGNOSTIC PRINT
C
      IF ( DFLG ) WRITE(DIAGNO,19) J,XNEW,K0,K1,F,DFDX,RATIO,ERR
C
C     CHECK ERROR FOR END OF BESSEL INVERSION
ↄ
```

```
      IF ( ERROR .LE. MXERR) THEN
      GO TO 100
      END IF
C
C     END OF INVERSION LOOP
C
      END DO
C
C     !!!!! END INVERSION !!!!!
C
C     INVERSION OF BESSEL FUNCTION HAS FAILED.
C     PRINT ERROR MESSAGE AND HALT!
C
      WRITE(OUT,20) Z,X,ERROR
C
C     JUMP TO END
C
      GO TO 200
C
C     INVERSION SUCCESSFUL, STORE VALUES AND GO TO NEXT POINT
C
  100 CONTINUE
C
      DXDZ     = ( XP - XM )/( ZP - ZM )
      RHELIX(I) = LHELIX*X/TWPI
      DRDZ(I)   = DXDZ/TWPI
      ZHELIX(I) = LHELIX*Z
C
C     CHECK ON CONICAL END TAPER EXTENSION
C
      IF ( INTFLG ) THEN
      RTEST     = RHELIX(I) - DRDZ(I)*ZHELIX(I)
      IF ( RTEST .GE. RINTCP(KTEST) ) THEN
      RCONE(KTEST)   = RHELIX(I)
      ZCONE(KTEST)   = ZHELIX(I)
      ICONE(KTEST)   = I
      SLOPE(KTEST)   = DRDZ(I)
      INTCPT(KTEST)  = RTEST
      WRITE(OUT,21) I,RHELIX(I),ZHELIX(I),DRDZ(I),RTEST
C
      KTEST = KTEST + 1
      IF ( KTEST .GT. KINTCP ) INTFLG = .FALSE.
      END IF
C
      END IF
C
C     CONICAL TAPER CHECK FINISHED
C     RETURN TO TOP OF POINTS LOOP AND SELECT NEXT POINT
C
      END DO
C
C     !!!!! END POINTS !!!!!
C
C
C     ALL POINTS CALCULATED, WRITE OUT VALUES
C
C
C
C     OUTPUT DATA FOR RUN TIME.
C
      RUN_END  = SECNDS( RUN_START )
      TICK_END = FLOAT( TickCount - TICK_START )/60.0
C
      WRITE(LISA,39) RUN_DATE, RUN_TIME
      WRITE(LISA,40) RUN_END,  TICK_END
C
      RUN_START = RUN_END
C
      DO J=1,NPTS
C
      I = NPTS + 1 - J
      WRITE(OUT,22) J,RHELIX(I),ZHELIX(I),DRDZ(I),J
```

```
      M = MOD(I-1,NLISA)
      IF ( M .EQ. 0 ) THEN
      WRITE(LISA,23) ZHELIX(I),RHELIX(I),DRDZ(I)
      END IF
C
      END DO
C
C     OUTPUT HELIX RADIAL PROFILE WITH THE CONE EXTENSION OPTIONS
C
C     SELECT CONE VALUES
C
      IF ( CNFLG ) THEN
C
      ZO  = ZTAPER - FLOAT( NPTS )*DZ
      DO K=1,KINTCP
C
      COUT  = CONE + K - 1
      B     = INTCPT(K)
      S     = SLOPE(K)
      RC    = RCONE(K)
      ZC    = ZCONE(K)
C
C     WRITE OUT POINTS
C
      Z   = 0.0
      WRITE(COUT,23) B,Z,S
      WRITE(LISA,23) Z,B,S
C
      DO J=1,NPTS
C
      I = NPTS - J + 1
      Z = ZO + FLOAT( J-1 )*DZ
      Z = LHELIX*Z
      M = MOD( J-1,NLISA )
C
      IF ( Z .LT. ZC ) THEN
C
      R = S*Z + B
      WRITE(COUT,23) R,Z,S
      IF ( M .EQ. 0 ) WRITE(LISA,23) Z,R,S
C
      ELSE
      WRITE(COUT,23) RHELIX(I),ZHELIX(I),DRDZ(I)
      IF ( M .EQ. 0 ) WRITE(LISA,23) ZHELIX(I),RHELIX(I),DRDZ(I)
      END IF
C
C     END J DO LOOP
C
      END DO
C
C     END K DO LOOP
C
      END DO
C
C     END CONE TAPER OUTPUT
C
      END IF
C
C==================================================================C
C                                                                  C
C >>>>>>>>>>>>>>>>>> BEGIN INTERACTIVE CALCULATIONS <<<<<<<<<<<<<<<<<<< C
C                                                                  C
C==================================================================C
C
C
C      INTERACTIVE CALCULATION OF HELIX OPERATING CHARACTERISTICS.
C
      IF ( .NOT. INTERACTIVE ) GO TO 200
C
C     CALCULATE TOTAL NUMBER OF POINTS FOR ARC LENGTH
C     INTEGRATION OF HELIX TAPER. READ TAPER NUMBER AND
C     LOAD R AND Z VALUES IN TO NEW ARRAYS.
```

```
C
      IPTS = INT( ZTAPER/DZ + 0.0001 ) + 1
      IF ( IPTS .LT. NPTS+2 ) IPTS = NPTS + 2
      IF ( IPTS .GT. MXPTS ) THEN
      WRITE( *,24) IPTS,MXPTS
      GO TO 200
      END IF
C
C     QUESTION USER TO BEGIN INTERACTIVE SESSION.
C
  110 CONTINUE
      WRITE( *,'(''  BEGIN INTERACTIVE HELIX CALCULATIONS? (Y/N)'')')
      READ( *,'(A)')  GO
      IF ( (GO .EQ. 'N' ) .OR. ( GO .EQ. 'n' ) ) GO TO 200
C
C     QUERY FOR SELECTION OF TAPER TO WORK WITH.
C
  120 CONTINUE
      I = C
      WRITE( *,25) KINTCP
      READ( *, *)  I
      IF ( ( I .LT. 0 ) .OR. ( I .GT. KINTCP ) ) THEN
      WRITE( *,26) I
      GO TO 120
      END IF
C
      ZARC(IPTS)     = ZTAPER*LHELIX
      RARC(IPTS)     = RU
      R_PRIME(IPTS) = 0.0
C
      IF ( I .NE. 0 ) THEN
      JSTOP = ICONE(I)
      ELSE
      JSTOP = NPTS
      END IF
C
      K = IPTS - 1
      DO J=1,JSTOP
C
      ZARC(K)     = ZHELIX(J)
      RARC(K)     = RHELIX(J)
      R_PRIME(K) = DRDZ(J)
      K           = K - 1
      END DO
C
      IF ( I .NE. 0 ) THEN
C
      S    = SLOPE(I)
      B    = INTCPT(I)
      Z    = ZHELIX(JSTOP)
      JPTS = INT( Z/(DZ*LHELIX) + 0.0001 )
      ZDIF = Z - DZ*LHELIX*FLOAT( JPTS )
C
      ELSE
C
      Z    = ZHELIX(NPTS)
      S    = DRDZ(NPTS)
      B    = RHELIX(NPTS) - DRDZ(NPTS)*Z
      JPTS = INT( Z/(DZ*LHELIX) + 0.0001 )
      ZDIF = Z - DZ*LHELIX*FLOAT( JPTS )
C
      END IF
C
      DO J=1,JPTS
      ZARC(K)     = Z - DZ*LHELIX*FLOAT( J )
      RARC(K)     = S*ZARC(J) + B
      R_PRIME(K) = S
      K           = K - 1
      END DO
C
      IF ( ABS( ZDIF ) .GT. 0.0001*DZ*LHELIX ) THEN
      ZARC(K)     = 0.0
```

```
      RARC(K)     = B
      R_PRIME(K) = S
      K           = K - 1
      END IF
C
      IF ( K .GT. 0 ) THEN
      IPTS = IPTS - K
      DO J=1,IPTS
      ZARC(J)     = ZARC(J+K)
      RARC(J)     = RARC(J+K)
      R_PRIME(J) = R_PRIME(J+K)
      END DO
      END IF
C
C     BASE HELIX VALUES HAVE BEEN LOADED INTO ARRAYS.
C     READ IN DESIGN INFORMATION AND CALCULATE OPERATING
C     CHARACTERISTICS.
C
C
C     RETURN HERE TO CONTINUE CALCULATIONS WITH SAME CONE
C
  130 CONTINUE
      WRITE( *,27)
      READ( *, *)  TEMP,RHO,ALPHA
  140 CONTINUE
      WRITE( *,28)
      READ( *, *)  N_UNIFORM,L_FRACT
  150 CONTINUE
      WRITE( *,29)
      READ( *, *)  B_PERP,I_SUPPLY,V_SUPPLY
  160 CONTINUE
      WRITE( *,30)
      READ( *, *)  D11,D22,T_INSULATE
C
C     CONVERT WIRE DIMENSIONS FROM MILS TO CM
C
      D1           = MIL_TO_CM*D11
      D2           = MIL_TO_CM*D22
      T_INSULATION = MIL_TO_CM*T_INSULATE
C
      IF (( I_SUPPLY .NE. 0.0 ) .AND. ( V_SUPPLY .NE. 0.0 )) THEN
      SUPPLY = .TRUE.
      END IF
C
      IF ( D2 .EQ. 0.0 ) THEN
      AREA    = PI*(D1/2.0)**2
      DELTA_R = D1 + 2*T_INSULATION
      ELSE
      AREA    = D1*D2
      DELTA_R = D2 + 2*T_INSULATION
      END IF
C
C     CALCULATE PITCH ANGLE CORRECTION FACTOR FOR USE IN COIL
C     CALCULATIONS, ie NUMBER OF WIRES AND CENTER-TO-CENTER
C     DISTANCE. NOTE:
C
C                      THETA = ARCTAN(2PI*R/LHELIX)
C       CORRECTION FACTOR = 1/SIN(THETA)
C
      F_THETA = SQRT( LHELIX**2 + (TWPI*RU)**2 )/(TWPI*RU)
C
C     NOTE THAT THE FACTOR OF 2 IN CALCULATING THE NUMBER OF
C     WIRES IN ONE LAYER OF THE BIFILAR HELIX (NWIRES) TAKES
C     INTO ACCOUNT BOTH COILS OF THE HELIX.
C
      N       = INT( L_FRACT*LHELIX/(F_THETA*( D1 + 2*T_INSULATION) )
     &             + 0.25 )
      NWIRES = 2*N
C
      RESIST = RHO*( 1 + ALPHA*( TEMP - 20 ) )/AREA
C
      KW2 = KHELIX**2
```

```
C
      DETA = ( D1 + 2*T_INSULATION )*F_THETA*KHELIX
      ETA1 = ( -FLOAT( N )*DETA + DETA )/2.0
      SUM_CETA = 0.0
      DO J=1,N
      ETA = ETA1 + FLOAT( J-1 )*DETA
      SUM_CETA = SUM_CETA + COS( ETA )
      END DO
      SUM_CETA = 0.4*KHELIX*SUM_CETA
C
      DO J=1,MXLAYERS
C
      DR = DELTA_R*FLOAT( J-1 )
      R  = RU + DR
      DO K=1,IPTS
      RHELIX(K) = RARC(K) + DR
      END DO
C
      X        = KHELIX*R
      X_KPRIME = X*KBES(X,1,-1)
      IF ( J .NE. 1 ) THEN
      CN(J) = CN(J-1) + SUM_CETA*ABS( X_KPRIME )
      ELSE
      CN(J) = SUM_CETA*ABS( X_KPRIME )
      END IF
C
      SUM1 = SQRT( 1.0 + KW2*RHELIX(1)**2 + R_PRIME(1)**2 )
      SUM2 = SQRT( 1.0 + KW2*RHELIX(2)**2 + R_PRIME(2)**2 )
      SUM3 = SQRT( 1.0 + KW2*RHELIX(IPTS)**2 + R_PRIME(IPTS)**2 )
      SUM  = ( ZARC(2) - ZARC(1) )*( SUM1 + SUM2 )/2.0
      SUM  = SUM + ( SUM2 + SUM3 )*DZ*LHELIX/2.0
C
      DO K=3,IPTS-1
      SUM = SUM+DZ*LHELIX*SQRT(1.0+KW2*RHELIX(K)**2+R_PRIME(K)**2)
      END DO
C
      L_TAPER    = SUM
      L_UNIFORM  = N_UNIFORM*LHELIX*SQRT( 1.0 + KW2*R**2 )
      L_LOOP     = PI*( RHELIX(1) + DR )
      L1_WIRE(J) = L_TAPER + L_LOOP + L_UNIFORM
      L2_WIRE(J) = 2.0*L_TAPER + 2.0*L_LOOP + L_UNIFORM
      L1_LAYER(J) = FLOAT( NWIRES )*L1_WIRE(J)
      L2_LAYER(J) = FLOAT( NWIRES )*L2_WIRE(J)
C
      IF ( J .GT. 1 ) THEN
      L1_TOTAL(J) = L1_TOTAL(J-1) + L1_LAYER(J)
      L2_TOTAL(J) = L2_TOTAL(J-1) + L2_LAYER(J)
      ELSE
      L1_TOTAL(J) = L1_LAYER(J)
      L2_TOTAL(J) = L2_LAYER(J)
      END IF
C
      R1_TOTAL(J) = RESIST*L1_TOTAL(J)
      R2_TOTAL(J) = RESIST*L2_TOTAL(J)
C
      CUR         = B_PERP/CN(J)
      IF ( SUPPLY ) THEN
            IF ( CUR .GT. I_SUPPLY ) CUR = I_SUPPLY
      END IF
C
      V1          = R1_TOTAL(J)*CUR
      V2          = R2_TOTAL(J)*CUR
C
      IF ( SUPPLY ) THEN
            IF ( V1 .GT. V_SUPPLY ) THEN
            VOLT1(J) = V_SUPPLY
            CURRENT1(J) = V_SUPPLY/R1_TOTAL(J)
            BFIELD1(J) = CN(J)*CURRENT1(J)
            ELSE
            VOLT1(J) = V1
            CURRENT1(J) = CUR
            BFIELD1(J) = CN(J)*CURRENT1(J)
```

```
            END IF
C
            IF ( V2 .GT. V_SUPPL! ) THEN
            VOLT2(J) = V_SUPPLY
            CURRENT2(J) = V_SUPPLY/R2_TOTAL(J)
            BFIELD2(J) = CN(J)*CURRENT2(J)
            ELSE
            VOLT2(J) = V2
            CURRENT2(J) = CUR
            BFIELD2(J) = CN(J)*CURRENT2(J)
            END IF
C
      ELSE
C
            VOLT1(J)    = V1
            VOLT2(J)    = V2
            CURRENT1(J) = CUR
            CURRENT2(J) = CUR
C
      END IF
C
      P1KW(J)     = CURRENT1(J)*VOLT1(J)/1000.0
      P2KW(J)     = CURRENT2(J)*VOLT2(J)/1000.0
C
C     END DO LOOP OVER HELIX LAYERS.
C     PRINT INFORMATION AND CHECK FOR
C     ADDITIONAL CALCULATIONS.
C
      END DO
C
      WRITE(OUT,31) ZTAPER,N_UNIFORM,L_FRACT
      WRITE(OUT,32) NWIRES,D1,D2,T_INSULATION
      WRITE(OUT,33) RHO,ALPHA,TEMP
      IF ( .NOT. SUPPLY) THEN
      WRITE(OUT,34) B_PERP
      ELSE
      WRITE(OUT,35) I_SUPPLY,V_SUPPLY
      END IF
C
      IF ( SUPPLY ) THEN
C
C     DEBUG DUMP TO SPOT OUTPUT PROBLEMS
C
      DUMP CN,CURRENT1,BFIELD1,L1_LAYER,R1_TOTAL,VOLT1,P1KW
C
      WRITE(OUT,36)
      DO J=1,MXLAYERS
      WRITE(OUT,38) J,CN(J),CURRENT1(J),BFIELD1(J),L1_LAYER(J)
     &              ,R1_TOTAL(J),VOLT1(J),P1KW(J)
      END DO
C
      WRI`.(OUT,37)
      DO J=!,MXLAYERS
      WRITE(OUT,38) J,CN(J),CURRENT2(J),BFIELD2(J),L2_LAYER(J)
     &              ,R2_TOTAL(J),VOLT2(J),P2KW(J)
      END DO
C
      ELSE
C
      WRITE(OUT,36)
      DO J=1,MXLAYERS
      WRITE(OUT,38) J,CN(J),CURRENT1(J),B_PERP,L1_LAYER(J)
     &              ,R1_TOTAL(J),VOLT1(J),P1KW(J)
      END DO
C
      WRITE(OUT,37)
      DO J=1,MXLAYERS
      WRITE(OUT,38) J,CN(J),CURRENT2(J),B_PERP,L2_LAYER(J)
     &              ,R2_TOTAL(J),VOLT2(J),P2KW(J)
      END DO
C
      END IF
```

```
C
C     PRINT INFORMATION TO TERMINAL
C
      WRITE( *,31) ZTAPER,N_UNIFORM,L_FRACT
      WRITE( *,32) NWIRES,D1,D2,T_INSULATION
      WRITE( *,33) RHO,ALPHA,TEMP
      IF ( .NOT. SUPPLY) THEN
      WRITE( *,34) B_PERP
      ELSE
      WRITE( *,35) I_SUPPLY,V_SUPPLY
      END IF
C
C SET PAUSE USING READ
C
      WRITE( *,'('' PRESS RETURN TO CONTINUE.'')')
      READ( *, *)
C
      IF ( SUPPLY ) THEN
C
      WRITE( *,36)
      DO J=1,MXLAYERS
      WRITE( *,38) J,CN(J),CURRENT1(J),BFIELD1(J),L1_LAYER(J)
     &             ,R1_TOTAL(J),VOLT1(J),P1KW(J)
      END DO
      WRITE( *,'('' PRESS RETURN TO CONTINUE.'')')
      READ( *, *)
C
      WRITE( *,37)
      DO J=1,MXLAYERS
      WRITE( *,38) J,CN(J),CURRENT2(J),BFIELD2(J),L2_LAYER(J)
     &             ,R2_TOTAL(J),VOLT2(J),P2KW(J)
      END DO
      WRITE( *,'('' PRESS RETURN TO CONTINUE.'')')
      READ( *, *)
C
      ELSE
C
      WRITE( *,36)
      DO J=1,MXLAYERS
      WRITE( *,38) J,CN(J),CURRENT1(J),B_PERP,L1_LAYER(J)
     &             ,R1_TOTAL(J),VOLT1(J),P1KW(J)
      END DO
      WRITE( *,'('' PRESS RETURN TO CONTINUE.'')')
      READ( *, *)
C
      WRITE( *,37)
      DO J=1,MXLAYERS
      WRITE( *,38) J,CN(J),CURRENT2(J),B_PERP,L2_LAYER(J)
     &             ,R2_TOTAL(J),VOLT2(J),P2KW(J)
      END DO
      WRITE( *,'('' PRESS RETURN TO CONTINUE.'')')
      READ( *, *)
C
      END IF
C
C
C     CHECK OPTIONS FOR NEW CALCULATIONS.
C
      WRITE( *,'('' CONTINUE INTERACTIVE HELIX CALCULATIONS?(Y/N)'')')
C
      READ( *,'(A)')  GO
C
      IF ( (GO .EQ. 'N' ) .OR. ( GO .EQ. 'n' ) ) GO TO 200
C
      WRITE( *,'(''     CALCULATE WITH NEW PARAMETERS: I_SELECT'')')
      WRITE( *,'(''    ------------------------------------------'')')
      WRITE( *,'(''            NEW TAPER CONE ----------- 1'')')
      WRITE( *,'(''            .........................        '')')
      WRITE( *,'(''            NEW WIRE MATERIAL -------- 2'')')
      WRITE( *,'(''            ...................  ..          '')')
      WRITE( *,'(''            NEW HELIX PARAMETERS ---- 3'')')
      WRITE( *,'(''            .........................        '')')
```

```
      WRITE( *,'(''             NEW POWER SUPPLY -------- 4'')')
      WRITE( *,'(''             ..........................      '')')
      WRITE( *,'(''             NEW WIRE DIMENSIONS ----- 5'')')
      WRITE( *,'(''             ---------------------------     '')')
      WRITE( *,'(''                             I_SELECT  =  '',$)')
C
      READ( *,'(I1)')  I_SELECT
C
      GO TO (120,130,140,150,160),I_SELECT
C
C
C
C     PROGRAM INVHLX FINISHED, HALT EXECUTION
C
  200 CONTINUE
C
C     CLOSE FILES BEFORE HALT OF EXECUTION
C
      CLOSE( UNIT = 10, STATUS = 'KEEP' )
      CLOSE( UNIT = 11, STATUS = 'KEEP' )
      CLOSE( UNIT = 15, STATUS = 'KEEP' )
      CLOSE( UNIT = 20, STATUS = 'KEEP' )
      CLOSE( UNIT = 21, STATUS = 'KEEP' )
      CLOSE( UNIT =  7, STATUS = 'KEEP' )
C
      STOP
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
                   FUNCTION KBES( ARG, N, KBOPT )
C
C***********************************************************************
C*                                                                     *
C*                                                                     *
C*                          FUNCTION KBES                              *
C*                                                                     *
C*                                                                     *
C***********************************************************************
C
      PARAMETER (NC=7)
C
      REAL KBES
C
      DIMENSION C1K0(NC),C1K1(NC),C2K0(NC),C2I0(NC),C2K1(NC)
     D          ,C2I1(NC)
C
C
C  SET COEFFICIENTS FOR THE CALCULATION OF THE K-ZERO
C  AND THE K-ONE MODIFIED BESSEL FUNCTIONS. THE FUNCTIONS
C  ARE EVALUATED USING POWER SERIES AND THE I-ZERO AND
C  THE I-ONE MODIFIED BESSEL FUNCTIONS. THE COEFFICIENTS
C  ARE TAKEN FROM " HANDBOOK OF MATHEMATICAL FUNCTIONS "
C  BY ABRAMOWITZ AND STEGUN, PAGES 378-379, NINTH DOVER
C  PRINTING. DIFFERENT SETS OF COEFFICIENTS ARE USED WHEN
C  THE ARGUMENT IS ABOVE OR BELOW 2.0.
C
C
C  COEFFICIENTS FOR K-ZERO AND K-ONE WHEN X .GE. 2.0.
C
      C1K0(1) =  1.25331414
      C1K0(2) = -0.07832358
      C1K0(3) =  0.02189568
      C1K0(4) = -0.01062446
      C1K0(5) =  0.00587872
      C1K0(6) = -0.00251540
      C1K0(7) =  0.00053208
C
      C1K1(1) =  1.25331414
      C1K1(2) =  0.23498619
      C1K1(3) = -0.03655620
```

```
      C1K1(4) =  0.01504268
      C1K1(5) = -0.00780353
      C1K1(6) =  0.00325614
      C1K1(7) = -0.00068245
C
C
C  COEFFICIENTS FOR K-ZERO, K-ONE, I-ZERO, AND I-ONE
C  FOR 0 .LT. X .LT. 2.0.
C
      C2K0(1) = -0.57721566
      C2K0(2) =  0.42278420
      C2K0(3) =  0.23069756
      C2K0(4) =  0.03488590
      C2K0(5) =  0.00262698
      C2K0(6) =  0.00010750
      C2K0(7) =  0.00000740
C
      C2I0(1) =  1.00000000
      C2I0(2) =  3.51562290
      C2I0(3) =  3.08994240
      C2I0(4) =  1.20674920
      C2I0(5) =  0.26597320
      C2I0(6) =  0.03607680
      C2I0(7) =  0.00458130
C
      C2K1(1) =  1.00000000
      C2K1(2) =  0.15443144
      C2K1(3) = -0.67278579
      C2K1(4) = -0.18156897
      C2K1(5) = -0.01919402
      C2K1(6) = -0.00110404
      C2K1(7) = -0.00004686
C
      C2I1(1) =  0.50000000
      C2I1(2) =  0.87890594
      C2I1(3) =  0.51498869
      C2I1(4) =  0.15084934
      C2I1(5) =  0.02658733
      C2I1(6) =  0.00301532
      C2I1(7) =  0.00032411
C
C
C  NOW CALCULATE K-ZERO AND K-ONE FRCM THE POWER SERIES.
C
C
      NC1 = NC + 1
C
      AST1 = 0.D0
      AST2 = 0.D0
      ARGI    = ARG
C
      IF ( ARGI .LT. 2.D0 ) GO TO 210
C
      A = 2.D0/ARGI
      DO 200 J=1,NC
      AST1 = AST1*A + C1K0(NC1-J)
      AST2 = AST2*A + C1K1(NC1-J)
  200 CONTINUE
      A = 1.D0/(SQRT( ARGI )*EXP( ARGI ) )
      AST1 = A*AST1
      AST2 = A*AST2
CC
      GO TO 290
C
  210 CONTINUE
      A = (ARGI/2.D0)**2
      A1 = (ARGI/3.75D0)**2
      A2 = 0.D0
      A3 = 0.D0
C
      DO 220 J=1,NC
      AST1 = AST1*A + C2K0(NC1-J)
```

```
      AST2 = AST2*A + C2K1(NC1-J)
      A2      = A2*A1 + C2I0(NC1-J)
      A3      = A3*A1 + C2I1(NC1-J)
  220 CONTINUE
      A4 = LOG( ARGI/2.D0 )
      AST1 = AST1 - A4*A2
      AST2 = AST2/ARGI + A4*A3*ARGI
C
  290 CONTINUE
C
C
C  NOW K-ZERO AND K-ONE HAVE BEEN CALCULATED FOR ALL
C  THE ARGUMENTS. NEXT CHECK TO SEE IF THE BESSEL FUNCTION (KBOPT= +1),
C  OR IT'S DERIVITIVE (KBOPT= -1) IS REQUESTED. THEN CHECK TO SEE IF
C  N IS 0, 1, OR .GT. 1.
C
      IF ( KBOPT .LT. 0 ) GO TO 400
C
C  FUNCTION VALUES ARE DESIRED. CHECK N.
C
      IF ( N-1 ) 300, 320, 340
C
  300 CONTINUE
      AK = AST1
C
      KBES = AK
C
      GO TO 1000
C
  320 CONTINUE
      AK = AST2
C
      KBES = AK
C
      GO TO 1000
C
  340 CONTINUE
      ANM1 = AST1
      AN   = AST2
      ARGI = ARG
C
      DO 350 J=2,N
      ANP1 = ANM1 + 2.D0*FLOAT(J-1)*AN/ARGI
      ANM1 = AN
      AN   = ANP1
  350 CONTINUE
      AK = ANP1
C
      KBES = AK
C
      GO TO 1000
C
C
C  K-N DERIVITIVES ARE REQUIRED.
C
  400 CONTINUE
C
      IF ( N-1 ) 410, 430, 450
C
  410 CONTINUE
      AK = -AST2
C
      KBES = AK
C
      GO TO 1000
C
  430 CONTINUE
      AK = -(AST1 + AST2/ARG )
C
      KBES = AK
C
      GO TO 1000
```

```
C
  450 CONTINUE
      ANM1 = AST1
      AN   = AST2
      ARGI = ARG
C
      DO 460 J=2,N
      ANP1 = ANM1 + 2.D0*FLOAT(J-1)*AN/ARGI
      ANM1 = AN
      AN   = ANP1
  460 CONTINUE
      AK = -( ANM1 + FLOAT(N)*ANP1/ARGI )
C
      KBES = AK
C
 1000 CONTINUE
C
      RETURN
C
      END
C
C
C
                         FUNCTION FTAPER( Z, ZTAPER, FOPT )
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                       C
C                                                                       C
C                                                                       C
C                      ************************                         C
C                      *                      *                         C
C                      *   FUNCTION SYNOPSIS  *                         C
C                      *                      *                         C
C                      ************************                         C
C                                                                       C
C                                                                       C
C-----------------------------------------------------------------------C
C                                                                       C
C                                                                       C
C                                                                       C
C                      *************************                        C
C                      *                       *                        C
C                      *   NUMERICAL METHODS   *                        C
C                      *                       *                        C
C                      *************************                        C
C                                                                       C
C                                                                       C
C                                                                       C
C-----------------------------------------------------------------------C
C                                                                       C
C                                                                       C
C                                                                       C
C                      *************************                        C
C                      *                       *                        C
C                      * VARIABLE DEFINITIONS  *                        C
C                      *                       *                        C
C                      *************************                        C
C                                                                       C
C                                                                       C
C                                                                       C
C-----------------------------------------------------------------------C
C                                                                       C
C                                                                       C
C                                                                       C
C                   ********************************                    C
C                   *                              *                    C
C                   * PARAMETERS,TYPES,DIMENSIONS  *                    C
C                   *                              *                    C
C                   ********************************                    C
C                                                                       C
      PARAMETER (MXCOEF=10)
C
```

```
      INTEGER      FOPT,OUT
C
      DIMENSION    CFUN(MXCOEF)
C                                                                       C
C=======================================================================C
C                                                                       C
C                                                                       C
C                                                                       C
C                             *************************                 C
C                             *                       *                 C
C                             *        COMMONS        *                 C
C                             *                       *                 C
C                             *************************                 C
C                                                                       C
C                                                                       C
      COMMON/FUNCTN/ NCOEF,CFUN
C                                                                       C
C=======================================================================C
C                                                                       C
C >>>>>>>>>>>>>>>>>>>>>> BEGIN FUNCTION FTAPER <<<<<<<<<<<<<<<<<<<<<<<< C
C                                                                       C
C=======================================================================C
C
C
      OUT    = 6
      PI     = 3.1415926535
      TWPI   = 2.0*PI
C
      FTAPER = 0.0
C
C     SELECT TYPE OF TAPER BASED ON FOPT VALUE
C
      GO TO ( 100, 200, 300 ), FOPT
C
C     VALUE OF FOPT IS NOT ALLOWED, RETURN.
C
      WRITE(OUT, 1) FOPT
    1 FORMAT('0','  FTAPER: FOPT IS OUT OF ALLOWED RANGE, FOPT = ',I3)
C
      FTAPER = 0.0
      RETURN
C
C     CUBIC SPLINE FIELD TAPER
C
  100 CONTINUE
C
      A = Z/ZTAPER
      FTAPER = ( -2.0*A + 3.0 )*A*A
      RETURN
C
C     SHIFTED COSINE FIELD TAPER
C
  200 CONTINUE
C
      A = Z/ZTAPER
      FTAPER = ( 1 - COS( PI*A ) )/2.0
      RETURN
C
C     POLYNOMIAL FIELD TAPER
C
  300 CONTINUE
C
      A = Z/ZTAPER
      S = CFUN( NCOEF )
      IEND = NCOEF - 1
      DO I=1,IEND
      S = S*A + CFUN( NCOEF - I )
      END DO
C
      FTAPER = S
      RETURN
C
```

```
C     END FUNCTION FTAPER
C
      END
```

```
      PROGRAM RIBTST2
C
C
      PARAMETER (PI=3.14159265359)
      COMMON/HLX1/ NLAYER,CURBAR,K,PHASE,ABAR
      REAL K,ABAR(20)
      DIMENSION XCENT(2),YCENT(2),ZCENT(2),IORIENT(2),PLEN(2),PWID(2)
      DIMENSION WMIN(2),WMAX(2),XY(2),AVG(2)
      CHARACTER ANS*1
C
      OPEN(10,FILE='RIBTST2.OUT')
      WRITE(6,*) 'ENTER HELIX PARAM; IN&OUT RADII,PERIOD,&WIDTH(CM)'
      ACCEPT *, AL,AU,P,W
      WRITE(6,*) 'ENTER NUMBER OF LAYERS'
      ACCEPT *,NLAYER
      WRITE(6,*) 'ENTER POWER SUPPLY CURRENT(A)'
      ACCEPT *,CUR
C
C     SCALING AND NORMALIZATION
C     CHANGE RADII TO METERS
C
      AL=AL*.01
      AU=AU*.01
      W=W*.01
      P=P*.01
      K=2.*PI/P
      PHASE=2.*PI*W/P
      DA=(AU-AL)/(NLAYER-1)
      CURBAR=(4.E-7)*CUR/W/NLAYER
      DO 10 I=1,NLAYER
   10 ABAR(I)=(AL+(I-1)*DA)*K
      WRITE(10,*)'AL=',AL,' AU=',AU,' P=',P,' W=',W
      WRITE(10,*)'NLAYER=',NLAYER,' CURR=',CUR
C
C     PROMPT FOR PROBE SPECIFICATION
C
      DO 140 I=1,2
C
  100 WRITE(6,*) 'ENTER COORDINATE (X,Y,Z CM) FOR CENTER OF PROBE ',I
      ACCEPT *,XCENT(I),YCENT(I),ZCENT(I)
  105 WRITE(6,*) 'ENTER PROBE ORIENTATION (1=XZ,2=YZ)'
      ACCEPT *,IORIENT(I)
      IF(IORIENT(I).NE.1 .AND. IORIENT(I).NE.2) GOTO 105
      WRITE(6,*) 'ENTER LENGTH AND WIDTH OF PROBE(CM) ',I
      ACCEPT *,PLEN(I),PWID(I)
      WRITE(6,110) I,XCENT(I),YCENT(I),ZCENT(I)
  110 FORMAT(1H1,1X,'CHARACTERISTICS OF PROBE ',I1,//,
     &       2X,'LOCATION OF CENTER OF PROBE:',/,5X,'X=',F10.5,/,
     &       5X,'Y=',F10.5,/,5X,'Z=',F10.5,/)
      IF(IORIENT(I).EQ.1) THEN
        WRITE(6,115)
  115   FORMAT(2X,'PROBE IS PARALLEL TO THE XZ PLANE',/)
      ELSE
        WRITE(6,120)
  120   FORMAT(2X,'PROBE IS PARALLEL TO THE YZ PLANE',/)
      ENDIF
      WRITE(6,125) PLEN(I),PWID(I)
  125 FORMAT(2X,'PROBE LENGTH=',F10.5,/,
     &       2X,'PROBE WIDTH= ',F10.5,///)
      WRITE(6,*) 'ARE THESE VALUES CORRECT(Y/N)?'
      READ(5,130) ANS
  130 FORMAT(A1)
      IF(ANS.NE.'Y' .AND. ANS.NE.'y') GOTO 100
C
C     WRITE TO OUTPUT FILE
C
      WRITE(10,110) I,XCENT(I),YCENT(I),ZCENT(I)
      IF(IORIENT(I).EQ.1) THEN
        WRITE(10,115)
      ELSE
        WRITE(10,120)
```

```
      ENDIF
       WRITE(10,125) PLEN(I),PWID(I)
      XCENT(I)=XCENT(I)*0.01
      YCENT(I)=YCENT(I)*0.01
      ZCENT(I)=ZCENT(I)*0.01
      PLEN(I)=PLEN(I)*0.01
       PWID(I)=PWID(I)*0.01
 140  CONTINUE
C
C   GET LENGTH OF AXIS AND STEP SIZE FOR Z
C
 145 WRITE(6,*) 'ENTER LENGTH OF Z-AXIS (CM)'
    ACCEPT *,ZLEN
     WRITE(6,*) 'ENTER STEP SIZE FOR Z (CM)'
    ACCEPT *,ZSTEP
    WRITE(6,150) ZLEN,ZSTEP
 150 FORMAT(/////,2X,'PROGRAM RUNS FOR A DISTANCE OF ',F10.5,' IN Z',/,
  &         2X,'WITH A STEP SIZE OF ',F10.5,//)
     WRITE(6,*) 'ARE THESE VALUES CORRECT(Y/N)?'
    READ(5,130) ANS
     IF(ANS.NE.'Y' .AND. ANS.NE.'y') GOTO 145
C
C   WRITE TO OUTPUT FILE
C
    WRITE(10,150) ZLEN,ZSTEP
    ZLEN=ZLEN*0.01
    ZSTEP=ZSTEP*0.01
C
C   COMPUTE EDGES OF PROBES
C
    WRITE(10,*) 'Z  PROBE1  PROBE2  BPERP'
    DO 160 I=1,2
      IF(IORIENT(I).EQ.1) THEN
        WMIN(I)=XCENT(I)-PWID(I)/2.0
        WMAX(I)=XCENT(I)+PWID(I)/2.0
       XY(I)=YCENT(I)
     ELSE
        WMIN(I)=YCENT(I)-PWID(I)/2.0
        WMAX(I)=YCENT(I)+PWID(I)/2.0
       XY(I)=XCENT(I)
     ENDIF
 160  CONTINUE
C
C   CALL ROUTINE FOR EACH STEP AND PRINT RESULTS FOR EACH
C
    LMAX=ZLEN/ZSTEP
    DO 200 I=1,LMAX
C
C     SET UP FOR EACH OF THE PROBES
C
     DO 175 IP=1,2
       ZMID=ZCENT(IP)+ZSTEP*FLOAT(I-1)
       ZMIN=ZMID-PLEN(IP)/2.0
       ZMAX=ZMID+PLEN(IP)/2.0
         CALL AVGVAL(WMIN(IP),WMAX(IP),ZMIN,ZMAX,XY(IP),IORIENT(IP),
  &          AVG(IP))
 175    CONTINUE
C
C     PRINT OUT RESULTS
C
     Z1=ZSTEP*FLOAT(I-1)
      BPERP=SQRT(AVG(1)**2+AVG(2)**2)
       WRITE(10,*) Z1,CHAR(9),AVG(1),CHAR(9),AVG(2),CHAR(9),BPERP
 200  CONTINUE
C
    END


     SUBROUTINE AVGVAL(WMIN,WMAX,ZMIN,ZMAX,XY,IORIENT,AVG)
C
C    THIS SUBROUTINE COMPUTES THE AVERAGE VALUE OF A FUNCTION OVER
C    THE AREA BOUNDED BY WMIN,ZMIN AND WMAX,ZMAX. THIS AVERAGE VALUE
```

```
C   IS FOUND BY DIVIDING THE INTEGRAL OF THE FUNCTION OVER THE AREA
C   BY THE AREA. THE INTEGRAL IS COMPUTED USING A TWO DIMENSIONAL
C   SIMPSON'S RULE ON INTERVALS OF THE AREA. THE SIZE OF THE INTERVAL
C   IS CHOSEN TO BE ONE HALF THE LENGTH OF THE SHORTER SIDE OF THE
C   RECTANGULAR AREA. FOR THIS REASON, THE LONGER EDGE SHOULD BE AN
C   INTEGER MULTIPLE OF ONE HALF THE SHORTER EDGE.
C
      PARAMETER (PI=3.14159265359)
      COMMON /HLX1/ NLAYER,CURBAR,K,PHASE,ABAR
      REAL K,ABAR(20)
      DIMENSION W(9,9)
      DATA ((W(I,J),J=1,9),I=1,9)
     & / 1.0, 4.0, 2.0, 4.0, 2.0, 4.0, 2.0, 4.0, 2.0,
     &    4.0,16.0, 8.0,16.0, 8.0,16.0, 8.0,16.0, 8.0,
     &    2.0, 8.0, 4.0, 8.0, 4.0, 8.0, 4.0, 8.0, 4.0,
     &    4.0,16.0, 8.0,16.0, 8.0,16.0, 8.0,16.0, 8.0,
     &    2.0, 8.0, 4.0, 8.0, 4.0, 8.0, 4.0, 8.0, 4.0,
     &    4.0,16.0, 8.0,16.0, 8.0,16.0, 8.0,16.0, 8.0,
     &    2.0, 8.0, 4.0, 8.0, 4.0, 8.0, 4.0, 8.0, 4.0,
     &    4.0,16.0, 8.0,16.0, 8.0,16.0, 8.0,16.0, 8.0,
     &    2.0, 8.0, 4.0, 8.0, 4.0, 8.0, 4.0, 8.0, 4.0 /
C
C   DETERMINE SIDE OF RECTANGLE AND LENGTH OF INTERVAL
C
      DISTW=WMAX-WMIN
      DISTZ=ZMAX-ZMIN
      IF(DISTW.GT.DISTZ) THEN
       H=DISTZ/4.0
       MAXJ=5
       MAXI=INT(DISTW/H)+1
      ELSE
       H=DISTW/4.0
       MAXI=5
       MAXJ=INT(DISTZ/H)+1
      ENDIF
C
C   COMPUTE INTEGRAL
C
      VOL=0.0
      DO 200 I=1,MAXI
       DO 100 J=1,MAXJ
        IF(IORIENT.EQ.1) THEN
C
C         PLATE PARALLEL TO XZ PLANE
C
        Y=XY
         X=WMIN+H*FLOAT(I-1)
        ELSE
C
C         PLATE PARALLEL TO YZ PLANE
C
        X=XY
         Y=WMIN+H*FLOAT(I-1)
        ENDIF
C
        R=SQRT(X**2+Y**2)
        IF(Y.EQ.0.0.AND.X.GT.0.0) THEN
          PHI=0.0
        ELSE IF(Y.EQ.0.0.AND.X.LT.0.0) THEN
          PHI=PI
        ELSE IF(Y.GT.0.0.AND.X.EQ.0.0) THEN
          PHI=PI/2.
        ELSE IF(Y.LT.0.0.AND.X.EQ.0.0) THEN
          PHI=3*PI/2.
        ELSE
          PHI=ATAN2(Y,X)
        ENDIF
        Z=ZMIN+H*FLOAT(J-1)
        RBARR=R*K
          CALL RIBHLX(RBARR,PHI,Z,BR,BPHI,BZ,BX,BY)
        if(r.eq.0.0.and.iorient.eq.2) then
        bx=-bx
```

```
      by=-by
      endif
       W1=W(I,J)
       IF(I.EQ.MAXI) W1=W1/2.0
       IF(J.EQ.MAXJ) W1=W1/2.0
       IF(IORIENT.EQ.1) THEN
         VOL=VOL+BY*W1
       ELSE
         VOL=VOL+BX*W1
       ENDIF
 100     CONTINUE
 200   CONTINUE
C
C   MAKE FINAL INTEGRAL CACULATION AND DIVIDE BY AREA FOR AVERAGE
C
     VOL=VOL*4.0*H**2/36.0
      AVG=VOL/(FLOAT(MAXI-1)*FLOAT(MAXJ-1)*H**2)
C
    RETURN
    END

C
      SUBROUTINE RIBHLX(RBARR,PHI,Z,BR,BPHI,BZ,BX,BY)
     COMMON/HLX1/ NLAYER,CURBAR,K,PHASE,ABAR
    COMMON/HLX2/ EPS,RBAR,RBAR2
C
     REAL K,ABAR(20),VALM(3)
C
    EPS=0.00000001
    RBAR=RBARR
     RBAR2=RBAR*RBAR
    PSI=PHI-K*Z
    PSI2=PSI+PHASE
    BR=0.
    BPHI=0.
    BZ=0.
C
    DO 30 I=1,NLAYER
      CALL SUMM(PSI,PSI2,ABAR(I),VALM)
      BR=BR+ABAR(I)*VALM(1)
       BPHI=BPHI+ABAR(I)*VALM(2)
 30    BZ=BZ+ABAR(I)*VALM(3)
C
     BR=-CURBAR*BR
     BPHI=-CURBAR*BPHI
     BZ=CURBAR*BZ
       BX=BR*COS(PHI)-BPHI*SIN(PHI)
       BY=BR*SIN(PHI)+BPHI*COS(PHI)
C
    RETURN
    END
C
     SUBROUTINE SUMM(PSI,PSI2,ABARI,VALM)
    COMMON/HLX2/ EPS,RBAR,RBAR2
      DIMENSION VALM(3),VALN(3),ERRORM(3),BM(3),BMO(3),RBR(3)
    LOGICAL*1 DONE,LFLG(3)
C
   M=0
    DONE=.FALSE.
    RBR(1)=1.
    RBR(2)=1.
    RBR(3)=RBAR
   DO 10 L=1,3
     BM(L)=0.0
     ERRORM(L)=1.
     LFLG(L)=.FALSE.
 10   CONTINUE
C
    DO WHILE (.NOT.DONE)
     M1=2*M+1
      CALL SUMN(M1,ABARI,VALN)
     DO 20 L=1,3
```

```
      IF(L.EQ.1) THEN
        ANGLE=-(SIN(M1*PSI2)-SIN(M1*PSI))/M1
      ELSE
        ANGLE=(COS(M1*PSI2)-COS(M1*PSI))/M1
      END IF
      IF(.NOT.LFLG(L)) THEN
        BMO(L)=BM(L)
        BM(L)=BM(L)+VALN(L)*RBR(L)*ANGLE
        IF(M1.NE.1) THEN
          ERRORM(L)=ABS(BMO(L)-BM(L))
        END IF
        IF(ERRORM(L).LT.EPS) THEN
          VALM(L)=BM(L)
          LFLG(L)=.TRUE.
        END IF
      END IF
 20   CONTINUE
      DONE=LFLG(1).AND.LFLG(2).AND.LFLG(3)
      M=M+1
      END DO
      RETURN
      END
C
      SUBROUTINE SUMN(M1,ABARI,VALN)
      COMMON/HLX2/ EPS,RBAR,RBAR2
      DIMENSION VAL(3),VALN(3),ERRORN(3),BJNM(3),BJNMO(3)
      LOGICAL*1 DONE,LFLG(3)
C
      N=0
      DONE=.FALSE.
      DO 10 L=1,3
        BJNM(L)=0.0
        ERRORN(L)=1.
        LFLG(L)=.FALSE.
 10   CONTINUE
C
      ABAR2=ABARI*ABARI
      RHO2=ABAR2+RBAR2
      RHO=SQRT(RHO2)
      RBRHO2=RBAR2/RHO2
      ABRHO2=ABAR2/RHO2
      XM=ABARI*RBAR/2./RHO
      XM2=XM*XM
C
      IF(M1.EQ.1) THEN
        FNM=1.
      ELSE
        FNM=XM**(M1-1)/FACTOR(M1)
      END IF
C
      DO WHILE (.NOT.DONE)
        CALL BMNTRM(M1,N,ABAR2,RHO,ABRHO2,RBRHO2,VAL)
        IF(N.NE.0) THEN
          FNM=FNM*XM2/N/(N+M1)
        END IF
        DO 20 L=1,3
          IF(.NOT.LFLG(L)) THEN
            BJNMO(L)=BJNM(L)
            BJNM(L)=BJNM(L)+FNM*VAL(L)
            IF(N.NE.0) THEN
              ERRORN(L)=ABS(BJNMO(L)-BJNM(L))
            END IF
            IF(ERRORN(L).LT.EPS) THEN
              VALN(L)=BJNM(L)
              LFLG(L)=.TRUE.
            END IF
          END IF
 20     CONTINUE
        DONE=LFLG(1).AND.LFLG(2).AND.LFLG(3)
        N=N+1
      END DO
      RETURN
```

```
      END
C
       SUBROUTINE BMNTRM(M1,N,ABAR2,RHO,ABRHO2,RBRHO2,VAL)
      DIMENSION VAL(3)
      N2=2*N
       CALL BESK(M1*RHO,M1+N2-1,BESS1,IER1)
       CALL BESK(M1*RHO,M1+N2,BESS2,IER2)
       IF(IER1.NE.0.OR.IER2.NE.0) THEN
         WRITE(6,*) 'BESSEL FUNCTION ERROR;M1,N2,RHO=',M1,N2,RHO
         WRITE(6,*) 'IER1,IER2=',IER1,IER2
        TIMF=TIMF/0.0
        STOP
      END IF
C br TERMS
       VAL(1)=((M1+N2)*BESS1+M1*BESS2/RHO)
C bphi TERMS
         VAL(2)=((1.-RBRHO2)*M1*BESS1+(1.-2.*RBRHO2)*(M1+N2)*BESS2/RHO)
C bz TERMS
         VAL(3)=(-ABRHO2*M1*BESS1+(1.-2.*ABRHO2)*(M1+N2)*BESS2/RHO)
      RETURN
      END
C
      FUNCTION FACTOR(N)
C   FACTORIAL FUNCTION
      REAL*8 FI,SUM
C
   11 FACTOR=1.
      IF(N-1)40,40,13
   13 IF(N-10)21,21,31
   21 DO 23 I=2,N
      FI=I
   23   FACTOR=FACTOR*FI
      GO TO 40
   31 SUM=0.
      DO 34 I=11,N
      FI=I
   34   SUM=SUM+DLOG(FI)
   35 FACTOR=3628800.*DEXP(SUM)
   40 RETURN
      END
C
C
C    ..................................................................
C
C    SUBROUTINE BESK
C
C    COMPUTE THE K BESSEL FUNCTION FOR A GIVEN ARGUMENT AND ORDER
C
C    USAGE
C         CALL BESK(X,N,BK,IER)
C
C    DESCRIPTION OF PARAMETERS
C         X  -THE ARGUMENT OF THE K BESSEL FUNCTION DESIRED
C         N  -THE ORDER OF THE K BESSEL FUNCTION DESIRED
C         BK -THE RESULTANT K BESSEL FUNCTION
C         IER-RESULTANT ERROR CODE WHERE
C           IER=0  NO ERROR
C           IER=1  N IS NEGATIVE
C           IER=2  X IS ZERO OR NEGATIVE
C           IER=3  X .GT. 60, MACHINE RANGE EXCEEDED
C           IER=4  BK .GT. 10**36
C
C    REMARKS
C         N MUST BE GREATER THAN OR EQUAL TO ZERO
C
C    SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C       NONE
C
C    METHOD
C         COMPUTES ZERO ORDER AND FIRST ORDER BESSEL FUNCTIONS USING
C         SERIES APPROXIMATIONS AND THEN COMPUTES N TH ORDER FUNCTION
C         USING RECURRENCE RELATION.
```

```
C       RECURRENCE RELATION AND POLYNOMIAL APPROXIMATION TECHNIQUE
C        AS DESCRIBED BY A.J.M.HITCHCOCK,'POLYNOMIAL APPROXIMATIONS
C       TO BESSEL FUNCTIONS OF ORDER ZERO AND ONE AND TO RELATED
C       FUNCTIONS', M.TRUE.A.C., V.11,1957,PP.86-88, AND G.N. WATSON,
C       'A TREATISE ON THE THEORY OF BESSEL FUNCTIONS', CAMBRIDGE
C       UNIVERSITY PRESS, 1958, P. 62
C
C
C ......... ...................................................
C
      SUBROUTINE BESK(X,N,BK,IER)
      DIMENSION T(12)
      BK=.0
      IF(N)10,11,11
   10 IER=1
      RETURN
   11 IF(X)12,12,20
   12 IER=2
      RETURN
C   THE LIMIT OF 60. MAY BE ABLE TO BE REVISED UPWARD
   20 IF(X-60.0)22,22 21
   21 IER=3
      RETURN
   22 IER=0
      IF(X-1.)36,36,25
   25 A=EXP(-X)
      B=1./X
      C=SQRT(B)
      T(1)=B
      DO 26 L=2,12
   26 T(L)=T(L-1)*B
      IF(N-1)27,29,27
C
C   COMPUTE KO USING POLYNOMIAL APPROXIMATION
C
   27 G0=A*(1.2533141-.1566642*T(1)+.08811128*T(2)-.09139095*T(3)
     & +.1344596*T(4)-.2299850*T(5)+.3792410*T(6)-.5247277*T(7)
     & +.5575368*T(8)-.4262633*T(9)+.2184518*T(10)-.06680977*T(11)
     & +.009189383*T(12))*C
      IF(N)20,28,29
   28 BK=G0
      RETURN
C
C   COMPUTE K1 USING POLYNOMIAL APPROXIMATION
C
   29 G1=A*(1.2533141+.4699927*T(1)-.1468583*T(2)+.1280427*T(3)
     & -.1736432*T(4)+.2847618*T(5)-.4594342*T(6)+.6283381*T(7)
     & -.6632295*T(8)+.5050239*T(9)-.2581304*T(10)+.07880001*T(11)
     & -.01082418*T(12))*C
      IF(N-1)20,30,31
   30 BK=G1
      RETURN
C
C   FROM KO,K1 COMPUTE KN USING RECURRENCE RELATION
C
   31 DO 35 J=2,N
      GJ=2.*(FLOAT(J)-1.)*G1/X+G0
      IF(GJ-1.0E36)33,33,32
   32 IER=4
      GO TO 34
   33 G0=G1
   35 G1=GJ
   34 BK=GJ
      RETURN
   36 B=X/2.
      A=.57721566+ALOG(B)
      C=B*B
      IF(N-1)37,43,37
C
C   COMPUTE KO USING SERIES EXPANSION
C
   37 G0=-A
```

```
      X2J=1.
      FACT=1.
      HJ=.0
      DO 40 J=1,6
      RJ=1./FLOAT(J)
      X2J=X2J*C
      FACT=FACT*RJ*RJ
      HJ=HJ+RJ
   40 G0=G0+X2J*FACT*(HJ-A)
      IF(N)43,42,43
   42 BK=G0
      RETURN
C
C     COMPUTE K1 USING SERIES EXPANSION
C
   43 X2J=B
      FACT=1.
      HJ=1.
      G1=1./X+X2J*(.5+A-HJ)
      DO 50 J=2,8
        X2J=X2J*C
        RJ=1./FLOAT(J)
        FACT=FACT*RJ*RJ
        HJ=HJ+RJ
   50   G1=G1+X2J*FACT*(.5+(A-HJ)*FLOAT(J))
      IF(N-1)31,52,31
   52 BK=G1
      RETURN
      END
```

REFERENCES

1. R. M. Phillips, IRE ED-7, 231 (1960)

2. R. M. Phillips, Proceedings of the 4th Int. Congress on Microwave Tubes, Scheveningen, 3-7 September, 1962, Centrex Pub. Co., Eindhoven, p 371.

3. C. E. Enderby and R. M. Phillips, IEEE Proc. 53, 1648 (1965).

4. H. P. Freund and A. K. Ganguly, Phys. Rev. A28, 3438 (1983).

5. A. K. Ganguly and H. P. Freund, Phys. Rev. A32, 2275 (1985).

6. A. K. Ganguly and H. P. Freund, Phys. Fluids 31,387(1988).

7. H. P. Freund and A. K. Ganguly, Phys. Rev. A34, 1242(1986).

8. Los Alamos Accelerator Code Group, "User's Guide for the POISSON/SUPERFISH Group of Codes," LA-UR-87-115, Los Alamos National Laboratory, Los Alamos, NM.

9. Los Alamos Accelerator Code Group, "Reference Manual for the POISSON/SUPERFISH Group of Codes," LA-UR-87-126, Los Alamos National Laboratory, Los Alamos, NM.

10. W. B. Hermannsfeldt, "Electron Trajectory Program," SLAC-226, Stanford Linear Accelerator Center, Stanford University, Stanford, CA, Nov. 1979.

11. G. Merdinian, G. Miram, R. Jackson, R. Parker, D. Pershing, "Ultralaminar 250 kV Gun for FastWave Devices," Technical Digest IEDM, 188 (1985).

12. N. Vanderplaats, private communication.

13. M. A. Heald and C. B. Wharton, Plasma Diagnostics with Microwaves, Robert E. Krieger Publishing Co., Huntington, N.Y., ©1965, pp. 322-4.

14. "A Turnstile Format Circular Waveguide Coupler," prepared for Mission Research Corporation by R. True, Electron Devices Div., Litton Systems, Inc., June 21, 1985.

15. D. Pershing, "Ubitron RF Output Monitor," Mission Research Corporation Report, MRC/WDCR-131, Apr. 1987.

16. R. E. Collin, Foundations of Microwave Engineering. ©1966 McGraw-Hill, Inc., Sects. 4.10, 4.11, 6.4.

17. G. Janzen and H. Stickel, "Mode Selective Directional Couplers for Overmoded Waveguide Systems," Int. J. Infrared and Millimeter Waves 5, 887 (1984).

18. M. S. Caceci and W. P. Cacheris, "Fitting Curve to Data," Byte, 340 (May 1984).

19. M. Sucher and J. Fox, eds., Handbook of Microwave Measurements, 3rd ed., Vol. I, H.J. Carlin, "Measurement of Power," ©1963 Polytechnic Press of the Polytechnic Institute of Brooklyn, pp.171-185.

20. S. Ramo, J.R. Whinnery, T. Van Duzer, Fields and Waves in Communication Electronics, 2nd ed., ©1984 John Wiley and Sons, pp. 279.

21. A. F. Harvey, Microwave Engineering ©1963 Academic Press, Inc. (London) Ltd., p. 248.

22. R. B. Dyott and M.C. Davies, "Interaction Between an Electron Beam of Periodically Varying Diameter and EM Waves in a Cylindrical Guide," IEEE Trans. Electron. Devices ED-13, 374 (1966).

23. N.A. Krall and A.W. Trivelpiece, *Principles of Plasma Physics* , (McGraw-Hill, New York, 1973), p.202.

24. S. Y. Park, J. M. Baird, R. A. Smith, and J. L. Hirshfield, J. Appl. Phys. 53, 1320 (1982).

25. M. C. Flythe, B. B. Amlicke, T. T. Fitzpatrick, "PRESCRIBE USER'S MANUAL," Mission Research Corporation Report MRC/WDC-R-136, May, 1987.